FRA5097
FREQUENCY RESPONSE ANALYZER

GPIB/USB INSTRUCTION MANUAL
Registered trademark

National Instruments is a registered trademark of National Instruments Corporation USA.

Other company names and product names are trademarks or registered trademarks of their respective owners.
INTRODUCTION

This instruction manual describes the FRA5097 GPIB and USB interface. Refer to the separate volume "FRA5097 INSTRUCTION MANUAL" for information on panel operations. The GPIB and USB interface of the FRA5097 has numerous functions, which allow control of nearly all front-panel operations. In addition, settings and measurement data can be read externally.

The FRA5097 GPIB/USB interface operation manual is comprised of the following sections.

1. Preparations Before Use
   Describes the procedures for setting addresses and delimiters of the GPIB, as well as the procedures for resetting the remote status.

2. Service Request and Status byte
   Details the service request (SRQ) and the status byte.

3. Program Messages
   Describes the formats for program codes, the list of commands, and individual commands.

4. Notes for Programming
   Describes cautionary instructions that are specific to the FRA5097, as well as notes for its use.

5. Sample Program
   Describes the sample program that uses the GPIB interface.

6. Specifications
   Describes the specifications for the GPIB and USB interface of the FRA5097.
1. Preparations before Use ..............................................................................................................1-1
   1.1 Overview of the FRA5097 GPIB/USB Interface .................................................................1-2
   1.2 USB Setup ..........................................................................................................................1-2
   1.3 GPIB Setup .........................................................................................................................1-2
   1.4 Interface Selection ..............................................................................................................1-3
   1.5 Setting GPIB Addresses .................................................................................................1-3
   1.6 Setting Delimiters .............................................................................................................1-4
   1.7 USB ID .................................................................................................................................1-4
   1.8 Resetting the Remote Status ............................................................................................1-5
   1.9 Notes ..................................................................................................................................1-5

2. Service Request and Status Byte .............................................................................................2-1
   2.1 Service Request ....................................................................................................................2-2
   2.2 Status Byte ..........................................................................................................................2-3

3. Program Messages ....................................................................................................................3-1
   3.1 Program Codes .......................................................................................................................3-2
      3.1.1 Setup Messages ............................................................................................................3-3
      3.1.2 Query Messages ...........................................................................................................3-3
      3.1.3 Parameters ....................................................................................................................3-3
      3.1.4 Input buffer ..................................................................................................................3-6
   3.2 Data Transfer .......................................................................................................................3-7
      3.2.1 Transfer of ASCII Format Data .................................................................................3-8
      3.2.2 Transfer of Binary Format Data .................................................................................3-9
   3.3 Program Code List ...............................................................................................................3-12
      3.3.1 Setup Message List ......................................................................................................3-13
      3.3.2 Query Message List .....................................................................................................3-16
   3.4 Individual program code description ...............................................................................3-19

4. Notes for Programming ............................................................................................................4-1

5. Sample Program ..........................................................................................................................5-1
   5.1 Overview of Sample Program ............................................................................................5-2
   5.2 Setup and Query ...................................................................................................................5-3
   5.3 Setting a Sweep Frequency Range and Sweep Measurement ...........................................5-5
Contents

5.4 Transfer of Measurement Data to Controller (1) ............................................ 5-7
5.5 Transfer of Measurement Data to Controller (2) ............................................ 5-9

6. Specifications ............................................................................................................ 6-1
   6.1 Interface Functions .................................................................................................. 6-2
   6.2 GPIB Bus Driver .................................................................................................... 6-2
   6.3 Applied Codes ...................................................................................................... 6-2
   6.4 Answers to Interface Messages ............................................................................ 6-3
   6.5 Multi-line Interface Messages .............................................................................. 6-4
Figures

Figure 1-1  Status Label ..........................................................1-5

Figure 3-1  Program Code Syntax ................................................3-2
Figure 3-2  Data Transfer Procedure ...........................................3-7

Tables

Table 2-1  Status Byte ..............................................................2-3

Table 3-1  Main Header List .......................................................3-12
Table 3-2  Setup Message List ....................................................3-13
Table 3-3  Query Message List ...................................................3-16
# 1. Preparations before Use

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Overview of the FRA5097 GPIB/USB Interface</td>
<td>1-2</td>
</tr>
<tr>
<td>1.2</td>
<td>USB Setup</td>
<td>1-2</td>
</tr>
<tr>
<td>1.3</td>
<td>GPIB Setup</td>
<td>1-2</td>
</tr>
<tr>
<td>1.4</td>
<td>Interface Selection</td>
<td>1-3</td>
</tr>
<tr>
<td>1.5</td>
<td>Setting GPIB Addresses</td>
<td>1-3</td>
</tr>
<tr>
<td>1.6</td>
<td>Setting Delimiters</td>
<td>1-4</td>
</tr>
<tr>
<td>1.7</td>
<td>USB ID</td>
<td>1-4</td>
</tr>
<tr>
<td>1.8</td>
<td>Resetting the Remote Status</td>
<td>1-5</td>
</tr>
<tr>
<td>1.9</td>
<td>Notes</td>
<td>1-5</td>
</tr>
</tbody>
</table>
1.1 Overview of the FRA5097 GPIB/USB Interface

The FRA5097 allows nearly all functions to be remotely set via the GPIB or USB. In addition, measurement data and setup conditions can be transferred externally, so that an automatic measurement system can be configured effortlessly.

1.2 USB Setup

The FRA5097 can be controlled by the USB Test and Measurement Class (USB-TMC) protocol. Most panel operations can be controlled, allowing the internal statuses, including setting values and errors, to be read.

- The USB connector is located on the rear panel.
- The USB interface is intended for use in a relatively favorable environment. Whenever possible, avoid use where the power supply may frequently fluctuate or be subject to excessive noise.

Install the USB-TMC class driver into the computer used for control and use a commercially available USB cable to connect the computer to the FRA5097. The driver installation file can be downloaded from the National Instruments Corporation website. The procedures for installing the driver are shown below.

1. Search for the VISA Run-time Engine page of the National Instruments Corporation website, or select “VISA driver downloads” at the following URL.
   http://www.ni.com/support/visa/

2. Download the VISA Run-time Engine from the VISA Run-time Engine page. User registration is required at this time. Download VISA Run-time Engine Version 3.3 or later.

3. The downloaded file is a self-extracting file. Extract the file and install it.

4. Completion of successful installation means that the USB-TMC class driver has been installed.

For details, visit the National Instruments Corporation website.

1.3 GPIB Setup

Mount a GPIB board (card) to the computer to control, using a commercially available GPIB cable. For details, refer to the user’s manual that comes with the GPIB controller board (card) to be used.
1.4 Interface Selection

Select an interface to use from GPIB or USB. The FRA5097 cannot be controlled through the deselected interface. The selected value is retained by the battery even after the power is turned off.

Note that GPIB is selected upon shipment.

Open menu [Output] [SELECT]; then, select [GPIB] or [USB] from the table using the ITEM keys.

--- CAUTION ---

Unexpected operation may occur in some computers if you switch the interface from USB to GPIB or unplug or plug the USB cable when the computer is recognizing the FRA5097.

1.5 Setting GPIB Addresses

Use the front panel to set a GPIB address of the FRA5097. Assign a unique value that differs from those allocated to the other components (computer etc.) which are connected via the GPIB cable.

The set value will be battery-protected even when the power is turned off.

The default address is "2."

Open the menu [Output][GPIB][ADDRESS], use the ENTRY keys to input a value between 0 and 30, and then press the ENTER key.
1.6 Setting Delimiters

Select GPIB delimiters that will be used for the FRA5097 to transmit data.

Use the front panel to make the settings. Either CR/LF or CR can be selected. An EOI signal will be output concurrently with the final byte of a delimiter. The set delimiter will be battery-protected even when the power is turned off. The default delimiter is "CR/LF^EOI."

Open the menu [Output][GPIB][OUTPUT DELIMITER]; then, select [CR/LF^EOI] or [CR^EOI] from the table using the ITEM keys.

The delimiter when USB is selected is fixed to "LF".

As a delimiter used for the FRA5097 to receive data, either <CR >, <LF> or an EOI signal or any combination of these can be accepted except for binary format data. Any delimiter that is used when the FRA5097 receives binary format data, will terminate a data transfer with a specified byte count or an EOI signal.

In this instruction manual, a "message terminator" or a "terminator" is referred to as a "delimiter."

1.7 USB ID

When more than one FRA5097 are USB connected within the system, use the following numbers for the applications to identify each FRA5097.

- Vendor ID : 3402(0x0D4A)
- Product ID : 10(0x0A)
- Serial number : Product production number (serial number)

The FRA5097 serial number is displayed when the menu [OUTPUT] [USB] is open. You cannot change the FRA5097 serial number.
1.8 Resetting the Remote Status

In the GPIB remote mode, REM is shown at the upper right of the LCD. See Fig. 1-1 Status display (reference).

![Status Display](image)

Figure 1-1 Status Label

In the remote status, all GPIB or USB operations will be enabled, and any key-in operation from the panel of the FRA5097 will be disabled except the LOCAL key. To reset the remote status, press the LOCAL key, set the REN line of the GPIB bus to High (False) via the controller, or issue the GTL command. However, the LOCAL key will be disabled as long as the FRA5097 is in the local lockout status. To reset the local lockout status, it is necessary to set the REN line of the GPIB bus to High (False) via the controller.

1.9 Notes

- Mate/unmate the GPIB connector in a condition in which all components on the bus are turned off.
- When using the GPIB, turn on all the components on the bus.
- In a system, up to 15 components can be connected to the GPIB, including the controller. As shown below, there are some restrictions on cable length.
  - The total extension of a cable must be 2 m x the number of components or 20 m - whichever is shorter.
  - The length of a single cable must be 4 m or shorter.
- Exercise particular care when setting GPIB addresses. If identical addresses are specified for different components in a system, they may be damaged.
- Exercise particular care when setting delimiters. Unless delimiters are set in a unified manner in a system, unexpected trouble may result.
- The GPIB and USB interfaces are intended for use in a relatively favorable environment. Whenever possible, avoid using the GPIB and USB in a place where the power supply may suffer frequent variations or it is subject to excessive noise.
2. Service Request and Status Byte

- 2.1 Service Request ..................................................... 2-2
- 2.2 Status Byte .......................................................... 2-3
2.1 Service Request

A service request (SRQ) is intended to set the SQR signal line of the bus lines to Low (True) and attempt an interrupt to the controller when the transmission of the SRQ is enabled and the FRA5097 is put in any of the following conditions:

- When a sweep is completed
- When one cycle of a manual sweep measurement is completed
- When overload input has taken place during a measurement
- When output data is ready for a query
- When some error has taken place

When an SRQ of the FRA5097 has been detected by the controller and a serial poll has been performed, the FRA5097 will turn down the SRQ after transferring the status byte (see the next section) to the controller.

Once permitted, the transmission of the SRQ will be run whether the status is remote or local.
2.2 Status Byte

The FRA5097 status byte contents are indicated in Table 2-1 Status Byte. Note that some reset conditions are different for GPIB and USB.

Table 2-1 Status Byte

<table>
<thead>
<tr>
<th>Bit</th>
<th>Content</th>
<th>Conditions for Setting to &quot;1&quot;</th>
<th>Conditions for Resetting to &quot;0&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>(MSB)7</td>
<td>0 (Unused)</td>
<td></td>
<td>Note 2 (Unused)</td>
</tr>
<tr>
<td>6</td>
<td>RQS</td>
<td>· When SRQ is transmitted</td>
<td>Note 1</td>
</tr>
<tr>
<td>5</td>
<td>Error occurred</td>
<td>· When an error has occurred</td>
<td>Note 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>· When an error code is read</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0 (Unused)</td>
<td></td>
<td>Note 2 (Unused)</td>
</tr>
<tr>
<td>3</td>
<td>Output ready</td>
<td>· When output data is ready for a query</td>
<td>Note 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>· When a subsequent query message is received</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Overvoltage input occurred</td>
<td>· When overvoltage input has occurred</td>
<td>Note 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>· When a normal measurement ends</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>· When a subsequent sweep is started</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>· When an error code is read</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Measure end</td>
<td>· When one cycle of a manual sweep measurement ends via GPIB</td>
<td>Note 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>· When a calibration ends</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sweep end</td>
<td>· When a sweep ends after reaching the maximum or minimum</td>
<td>Note 4</td>
</tr>
<tr>
<td>(LSB)0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Note 1**: · When a DCL or SDC (device clear) is received.
· After the status byte is read (due to a serial poll while an SRQ is being output or in response to a query message).

**Note 2**: · Constantly 0 because this is not used.

**Note 3**: · If this setting is made when output data is ready for a query, then the setting will remain even after the data has been output. The SRQ will be transmitted immediately if the transmission of the SRQ (service request) is then enabled for "Output ready" in response to the setup message "SRqenable 8".

**Note 4**: · When a device clear (DCL or SDC) signal is received.
· GPIB : After serial poll during the SRQ output, and also after reading the status byte as requested by a query message.
· USB : After reading the status byte as requested by a query message.
For USB, execution of a serial poll does not result in clearing.
3. Program Messages

3.1 Program Codes..................................................3-2
3.1.1 Setup Messages ..........................................3-3
3.1.2 Query Messages.......................................... 3-3
3.1.3 Parameters...................................................3-3
3.1.4 Input buffer ..................................................3-6
3.2 Data Transfer......................................................3-7
3.2.1 Transfer of ASCII Format Data....................3-8
3.2.2 Transfer of Binary Format Data..................3-9
3.3 Program Code List ..........................................3-12
3.3.1 Setup Message List................................... 3-13
3.3.2 Query Message List ..................................3-16
3.4 Individual program code description.............3-19
3.1 Program Codes

If a program code contains an unspecified header or parameter, the input buffer will be cleared when the program code is interpreted, and no subsequent program code will be executed.

A program code is composed of a header part and a parameter part that follows it, which are divided by ";" (semicolon). The program code syntax is as follows:

![Program Code Syntax Diagram]

**Figure 3-1 Program Code Syntax**

A header part is comprised of one main header and two or more sub headers whose number depends on the main header (however, a header part may contain optional sub headers or no sub header). The main header and sub headers are divided by at least one <TAB>, space or "," (comma).

**Ex:**
- osc offset <parameter part>
- sweep resolution log decade <parameter part>
- ?identifier No parameter contained

When a header part so requires, its parameter part is made up of the required number of parameters which are divided by ",", however a parameter part is generally optional. In addition, either the numerical format or string (mnemonic) format can be used for the parameter part.

**Ex:**
- sweep range <minimum>, <maximum>
- sweep range <minimum>, The maximum is optional
- sweep range <minimum> Same as above
- sweep range ,, <maximum> The minimum is optional
3.1 Program Codes

3.1.1 Setup Messages
Setup messages refer to the program codes that are intended to make a range of settings for the FRA5097 and instruct it to perform its functions.

3.1.2 Query Messages
Query messages refer to the program codes whose header parts are all prefixed with "?" (a question code). They are intended to run queries about the status, setups or measurement data of the FRA5097 and prompt the FRA5097 to output answer messages to them. Query messages generally have corresponding setup messages, except some exceptional ones. Any query message, which has the corresponding setup message, is represented by that setup message, except that its header part is prefixed with "?" (a question code).

When the FRA5097 has received a query message, it will get ready to respond to it by checking the relevant status, setups or measurement data. If the FRA5097 is then specified as a talker, it will output an answer message to the query. Any query message that has a corresponding setup message, follows the format of that setup message. Also, a header can be appended to a query message using the header setup message "SETUP HEADER ON". The status will become "SETUP HEADER OFF" by power on or when a DCL or SDC (device clear) is received.

There are setup messages for data such as measurement data to specify data formats. If a format is specified by a setup message, then the specified format will be used for that data. When more than one query is put to the FRA5097 at a time, the FRA5097 will receive only the final query and ignore all preceding queries. Also, when the FRA5097 receives a new query before completing the output to a previous query, it will start output to the new query.

3.1.3 Parameters
There are parameters in numeric and string formats.

a) Parameters in numeric format
According to individual setup items, the following three numeric formats are available for parameters used in setup messages or for answer messages to be output to query messages.

- NR1 format  Integer format
- NR2 format  Real number format
- NR3 format  Real number format (exponent format)

(Contd.)
3.1 Program Codes

- NR1 format
  The NR1 is an integer format (a format using numbers that do not include any decimal point. The decimal point of an integer is implicitly regarded as being placed at the end of its final digit).

  ±DDDD
  ◎At the time of input, any reading zero or space is ignored.
  ◎At the time of output, a reading zero is a space.
  ◎The sign is represented by "+" and "-".
  ◎If the sign is omitted at the time of input, it will be interpreted as "+". Any number of digits can be used.
  ◎At the time of output, "+" may be represented by a space according to the setup item. The fixed number of digits is used according to the setup item.

  Ex:
  +01234
  −500
  18

- NR2 format
  The NR2 is a real number format (a format using numbers that include decimal points, each of which is represented by "." (period)).

  ±DD.DD
  ◎At the time of input, any reading zero or space is ignored. If the numeric value before the decimal point is omitted, the integer part will be interpreted as 0.
  ◎At the time of output, a reading zero is a space.
  ◎The sign is represented by "+" and "-".
  ◎If the sign is omitted at the time of input, it will be interpreted as "+". Any number of digits can be used.
  ◎At the time of output, "+" may be represented by a space according to the setup item. The fixed number of digits is used according to the setup item.

  Ex:
  +012.34
  −50.0
  1.8

- NR3 format
  The NR3 is a real number (exponent) format.

  ±DD.DD E±DD
  ◎Exponent part
  ◎At the time of input, any reading zero or space is ignored. The sign is represented by "+" and "-". If the sign is omitted, it will be interpreted as "+". Any number of digits can be used.
  ◎At the time of output, a reading zero is "0" rather than a space. The numeric value is a multiple of 3 and is composed of "E" + Sign + 2-digit number, for which fixed 4 digits are used.
  ◎Same as NR2 format.

  Ex:
  +0.1234E +03
  −50.0E -06
  1.8E-9
To set a floating-point number as a parameter in a setup message, any of the three formats (NR1, NR2 and NR3) can be used.
To set an integral number as a parameter in a setup message, only the NR1 format can be used.
In actuality, a value that is set in a setup message will be rounded so that it will be within each parameter's resolution. Therefore, if a higher-digit number is specified, some digits that follow a certain number of digits will be discarded (rounded off). Even in this case, however, normal scaling will be performed for a decimal point or for exponents, if any.

b) Parameters in string format
To use data in string format as a parameter in a setup message or to transmit data in string format in response to a query message, the following formats are available:

- Mnemonic (string) format
- General string (title etc.) format

- Mnemonic (string) format
For parameters in NR1 or string format, the mnemonic (string) format can be substituted for the integer (NR1) format.
For reception, the FRA5097 can use either of the integer and mnemonic formats.
In mnemonic format, an intended string can be written as is. The mnemonic format is case-independent.
The FRA5097 can transmit in mnemonic format in response to the mnemonic setup message "SETUP MNEMONIC ON" Also, the FRA5097 will output in uppercase without any truncation.

Ex: OSCILLATOR MODE ON, Quick, zero
    DISPLAY ANALYSIS CH1BYCH2

(Contd.)
3.1 Program Codes

- General string (title etc.) format
  Follow the notation shown below for any other general string format. Note that any string output by the FRA5097 will be enclosed with ['"] (a double quotation).

  String data including a separator
  "

  String data excluding a separator
  \\

  Separator: Either SP, ",", or ":"
  SP: At least one <TAB> or space code
  ": Double quotation code
  ": Single quotation code
  "¥" or "\": "¥" for JIS and "\" for ASCII. These are used when ",", "¥" or "\" should be contained in a string.

  Ex:  data write title <tag number>, 'sample title'
    file load disk DATA.DAT, <tag number>
    data write title <tag number>, "include quote mark like as \, \ and ",",
    The string to be passed is: include quote mark like as"\,\ and'.
    data write title <tag number>, "these ; , aren't terminators."
    The string to be passed is: these ; , aren't terminators.

3.1.4 Input buffer

- You can send as many commands as possible at once, provided the total contents fit within the input buffer capacity (4096 bytes).
- The sent commands are first stored in the input buffer, then they are interpreted and executed one at a time.
- When an improper command found during an interpretation or an execution, an error is issued, and subsequent executions are aborted.
- Once the interpretations and executions are completed, the input buffer is cleared, and inputting of next command becomes possible.
- When sending a large size data (for instance a "Data Write data" command), divide such data to be below the buffer capacity before transferring.
3.2 Data Transfer

The FRA5097 can output/receive data to/from the controller using a data transfer message. The data transfer procedure is shown below.

Refer to "3.3 Program Code List" for details on program codes.

![Data Transfer Procedure Diagram]

- **Specifying the transfer format**
  The format of data to be transferred (binary or ASCII) and its configuration (types or combinations of frequency, gain, phase etc.) can be set by means of the setup message "data template"

- **Reading data (data output)**
  If the FRA5097 has been specified as a talker after receiving the query message "?data read" in which the number of data to be read is defined by a parameter, it will output the data to the controller.

- **Writing data (data input)**
  If the FRA5097 has received the setup message "data write" in which the number of data to be written is specified by a parameter, it will receive the data from the controller. The type of the written data is "OPERATED" which indicates that the data has undergone an arithmetic calculation.
3.2 Data Transfer

3.2.1 Transfer of ASCII Format Data
The FRA5097 will transfer data in ASCII format if the data transfer format is set to ASCII through the setup message "data template string".

![Diagram showing numeric format data and delimiter]

The measurement data in ASCII format refers to a series of numerical data blocks, of which the number and configuration are pre-defined. "," (commas) are used between individual data in a block. Interblock delimiters are as follows:

- At time of reception: Either <CR>, <LF> or EOI or any combination of these can be accepted.
- At time of transmission: The delimiter that was selected from the menu [Output][GPIB] [OUTPUT DELIMITER] will be used. However, no EOI is appended to an interblock delimiter.

The pre-defined number of blocks will be transferred.

Shown below are the I/O form and transfer format that are used at power-on or when a DCL or SDC (device clear) is received:

- Transfer format: ASCII format
- Per-block configuration: <frequency (Hz)>, <gain (dB)>, <phase (deg)>

<table>
<thead>
<tr>
<th>NR2</th>
<th>NR2</th>
<th>NR2</th>
</tr>
</thead>
<tbody>
<tr>
<td>17 characters</td>
<td>8 characters</td>
<td>7 characters</td>
</tr>
</tbody>
</table>

a) Example of ASCII format data transmission from FRA5097
- Setup message to FRA5097
  - data template string, sweep, logr, theta (delimiter)
  - Per-block configuration: <frequency>, <gain (log)>, <phase> in ASCII format
  - ?data read data 1,0,3 (delimiter)
  - Transfers 3 blocks of data starting at block 0 of tag 1

- Answer from FRA5097
  - 1000000.0000, 123.450, -134.23 (block delimiter)
  - 100000.0000, 0.012, 0.34 (block delimiter)
  - 1000.0000, -84.544, 140.33 (delimiter)
b) Example of ASCII format data reception by FRA5097

- Setup message and transfer data to FRA5097
  - data template string, 1, 5, 6 (delimiter)
  - Per-block configuration: <frequency>, <a>, <b> in ASCII format
  - data write data 1, 3, 2 (delimiter)

Write 2 blocks of data starting at block 3 of tag 1:
- 1.0E+6, 1.0, -1.000 (delimiter)
- 1.0E+6, -2.23, 2.34 (delimiter)

The data received will be written as operated data (OPERATED).

3.2.2 Transfer of Binary Format Data

When the data transfer format has been set to anything other than "string" via the setup message "data template" the FRA5097 will transfer the data in the specified binary format.

The measurement data in binary format refers to a series of floating-point data blocks in IEEE 754 (subset), of which the number and configuration are pre-defined. There is no delimiter between individual data in a block or between blocks.

No delimiter setting is required for the entire data, either. At the time of reception, a data transfer will terminate when a specified byte count or an EOI message is received, while at the time of transmission, a data transfer will terminate when an EOI is output together with the final byte of the entire data row.
c) Example of binary format (IEEE Double) data transmission from FRA5097

- Setup message to FRA5097
  
  data template double, sweep, logr, theta (delimiter)
  
  Per-block configuration: <frequency> <gain (log)> <phase> in IEEE Double
  
  The size of a single block is 8bytes x 3 = 24bytes
  
  data read data 2, 10, 200 (delimiter)
  
  Transfers 200 blocks of data starting at block 10 of tag 2

- Answer from FRA5097
  
  #504800 4,800bytes data row in IEEE Double format (delimiter)
  
  #: Indicates the binary data format.
  
  5: Indicates that the decimal digit string showing the byte count of the data is in five digits.
  
  04800: Indicates that the data is 4,800bytes (24bytes per block x 200 blocks).
  
  (delimiter): Indicates the delimiter that was selected from the menu [Output][GPIB][OUTPUT DELIMITER] (CR/LF^EOI or CR^EOI).

d) Example of binary format (IEEE Float (reverse byte order)) data reception by FRA5097

- Setup message and transfer data to FRA5097
  
  data template invfloat, sweep, r (delimiter)
  
  Per-block configuration: <frequency> <gain> in IEEE Float (reverse byte order)
  
  The size of a single block is 4bytes x 2 = 8bytes.
  
  data write 3, 100, 10 (delimiter)
  
  Writes 10 blocks of data starting at block 100 of tag 3
  
  #3080 80bytes data in IEEE Float format (reverse byte order)
  
  #: Indicates the binary data format.
  
  3: Indicates that the decimal string showing the byte count of the data is in 3 digits. An ASCII character between 1-9 can be specified.
  
  080: Indicates that the data is 80bytes (8bytes per block x 10 blocks).

An EOI is appended to the final data byte.

Even when the FRA5097 has received an EOI during data reception (when it has not completed receiving a specified byte count), it will terminate the reception of the binary data after receiving the bytes that have reached up to that time.
3.2 Data Transfer

e) IEEE floating point format

Through the setup message "data template", the following four IEEE floating point formats can be used as a transfer format for measurement data or operated data:

- IEEE double-precision floating point format (big endian)
- IEEE single-precision floating point format (big endian)
- IEEE double-precision floating point format (little endian)
- IEEE single-precision floating point format (little endian)

● IEEE double-precision floating point format in normal byte order (8 bytes per data)

```
Transfer order
1 2 3 4 5 6 7 8
s: Sign of mantissa 0: positive, 1: negative
e: Exponent part (11 bit) exp: 0 to 2,047
m: Mantissa (52 bit) mantissa
```

Numeric value = \((-1)^s \times 2^{(exp-1023)} \times (1+mantissa/2^{52})\)

However, when exp is 0, the underlined 1 is omitted

● IEEE single-precision floating point format in normal byte order (4 bytes per data)

```
Transfer order
1 2 3 4
s: Sign of mantissa 0: positive, 1: negative
e: Exponent part (8 bit) exp: 0 to 255
m: Mantissa (23 bit) mantissa
```

Numeric value = \((-1)^s \times 2^{(exp-127)} \times (1+mantissa/2^{23})\)

However, when exp is 0, the underlined 1 is omitted

● IEEE double-precision floating point format in reverse byte order

● IEEE single-precision floating point format in reverse byte order

Any of these formats is identical to the corresponding IEEE double-precision or single-precision floating-point format in normal byte order, except that the order of bytes to be transferred is reverse.
3.3 Program Code List

In a keyword string in headers or parameters that are used with the FRA5097, any number of characters that come before the final character can be omitted, except for some characters at the head of the string that may not be omitted. In the following description, characters that cannot be omitted will be represented in uppercase, and those which can be omitted will be represented in lowercase. A fully spelled header etc. makes the corresponding program code easier to understand, while an abbreviated one allows the controller resources or the transfer rate of a program code to be improved.

The main headers used in the FRA5097 program code are indicated in Table 3-1 Main Header List.

<table>
<thead>
<tr>
<th>Function</th>
<th>Main header</th>
<th>Action &amp; setup range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arithmetic calculation control</td>
<td>CAIculation</td>
<td>Arithmetic calculation, differentiation &amp; integration, open-to-close loop conversion</td>
</tr>
<tr>
<td>Amplitude compression setup</td>
<td>COMpression</td>
<td>A range of parameters for amplitude compression control</td>
</tr>
<tr>
<td>Data processing</td>
<td>DAta</td>
<td>Tag switching, data navigation, display switching, data exchange to/from controller, format</td>
</tr>
<tr>
<td>Display control</td>
<td>Display</td>
<td>Analysis mode, graph display control, display range, marker</td>
</tr>
<tr>
<td>File processing</td>
<td>File</td>
<td>USB flash drive, mass memory, permanent memory, file list, deletion, rename</td>
</tr>
<tr>
<td>Input part control</td>
<td>INput</td>
<td>Overload detection setup, equalization, weighting etc.</td>
</tr>
<tr>
<td>Measurement control</td>
<td>Measure</td>
<td>Integration control, delay control, measurement mode, harmonics etc.</td>
</tr>
<tr>
<td>Oscillator control</td>
<td>OSCillator</td>
<td>Oscillator control, such as frequencies, amplitudes, offset, phases and waveforms, ON/OFF</td>
</tr>
<tr>
<td>Setup proces-sing of varied types</td>
<td>SEtup</td>
<td>Setups of calibration, clock, initialization, buzzer, etc.</td>
</tr>
<tr>
<td>Others</td>
<td>Depends on the program code</td>
<td>SRQ mask, status byte, error, overload detection check etc.</td>
</tr>
</tbody>
</table>

In actuality, program codes are case-independent, so any of the following representations can be substituted for the oscillator control header "OScillator":

```
os
oScill
oscillator
```
### 3.3.1 Setup Message List

**Table 3-2 Setup Message List (contd.)**

<table>
<thead>
<tr>
<th>Function</th>
<th>Header</th>
<th>Ref. page</th>
<th>Query</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run messages for arithmetic calculations on data</td>
<td>CAIculatin Arithmetic</td>
<td>3-20</td>
<td>Present</td>
</tr>
<tr>
<td>Arithmetic calculation</td>
<td>CAIculatin Arithmetic</td>
<td>3-20</td>
<td>Present</td>
</tr>
<tr>
<td>Differentiation/integration</td>
<td>CAIculatin Jw</td>
<td>3-22</td>
<td>Present</td>
</tr>
<tr>
<td>Open-to-close loop conversion</td>
<td>CAIculatin Loop</td>
<td>3-24</td>
<td>Present</td>
</tr>
<tr>
<td>Setup messages for amplitude compression control</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reference amplitude level</td>
<td>CAmpression Amplitude</td>
<td>3-26</td>
<td>Present</td>
</tr>
<tr>
<td>Amplitude correction factor</td>
<td>CAmpression Correction</td>
<td>3-27</td>
<td>Present</td>
</tr>
<tr>
<td>Maximum error rate of reference amplitude</td>
<td>CAmpression Error</td>
<td>3-28</td>
<td>Present</td>
</tr>
<tr>
<td>Output amplitude limitation</td>
<td>CAmpression Limit</td>
<td>3-29</td>
<td>Present</td>
</tr>
<tr>
<td>Amplitude compression mode</td>
<td>CAmpression mode</td>
<td>3-30</td>
<td>Present</td>
</tr>
<tr>
<td>Maximum retry times of correction</td>
<td>CAmpression Retry</td>
<td>3-31</td>
<td>Present</td>
</tr>
<tr>
<td>Setup messages for data control</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tag-number switching</td>
<td>DAta CUrent</td>
<td>3-32</td>
<td>Present</td>
</tr>
<tr>
<td>Display data switching</td>
<td>DAta Display</td>
<td>3-33</td>
<td>Present</td>
</tr>
<tr>
<td>Entry of equalization data</td>
<td>DAta Equalize</td>
<td>3-34</td>
<td>Absent</td>
</tr>
<tr>
<td>Register in open equalizer memory</td>
<td>DAta Open</td>
<td>3-35</td>
<td>Absent</td>
</tr>
<tr>
<td>Register in short equalizer memory</td>
<td>DAta Short</td>
<td>3-43</td>
<td>Absent</td>
</tr>
<tr>
<td>Data transfer format</td>
<td>Data Template</td>
<td>3-44</td>
<td>Present</td>
</tr>
<tr>
<td>Data transfer from controller</td>
<td>DAta Write data</td>
<td>3-46</td>
<td>Absent</td>
</tr>
<tr>
<td>Transfer of general character strings from controller</td>
<td>DAta Write Title</td>
<td>3-47</td>
<td>Absent</td>
</tr>
<tr>
<td>Setup messages for display control</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Analysis mode</td>
<td>Display ANAlysis</td>
<td>3-48</td>
<td>Present</td>
</tr>
<tr>
<td>Autoscale</td>
<td>Display AUto</td>
<td>3-49</td>
<td>Present</td>
</tr>
<tr>
<td>Grid ON/OFF</td>
<td>Display Grid mode</td>
<td>3-50</td>
<td>Present</td>
</tr>
<tr>
<td>Grid type</td>
<td>Display Grid Type</td>
<td>3-51</td>
<td>Present</td>
</tr>
<tr>
<td>Grid mode</td>
<td>Display Grid Style</td>
<td>3-52</td>
<td>Present</td>
</tr>
<tr>
<td>Setup of active marker</td>
<td>Display MAker Active</td>
<td>3-54</td>
<td>Present</td>
</tr>
<tr>
<td>Setup of delta marker mode</td>
<td>Display MAker mode</td>
<td>3-55</td>
<td>Present</td>
</tr>
<tr>
<td>Graph display mode</td>
<td>Display mode</td>
<td>3-56</td>
<td>Present</td>
</tr>
<tr>
<td>Phase display range</td>
<td>Display Phase</td>
<td>3-60</td>
<td>Present</td>
</tr>
<tr>
<td>Display scale (X axis)</td>
<td>Display Scale Xaxis</td>
<td>3-61</td>
<td>Present</td>
</tr>
<tr>
<td>Display scale (Y&lt;sub&gt;-1&lt;/sub&gt; axis)</td>
<td>Display Scale Y1axis</td>
<td>3-62</td>
<td>Present</td>
</tr>
<tr>
<td>Display scale (Y&lt;sub&gt;-2&lt;/sub&gt; axis)</td>
<td>Display Scale Y2axis</td>
<td>3-63</td>
<td>Present</td>
</tr>
<tr>
<td>Display units setting</td>
<td>Display Units</td>
<td>3-64</td>
<td>Present</td>
</tr>
<tr>
<td>Graph display window mode</td>
<td>Display Window</td>
<td>3-65</td>
<td>Present</td>
</tr>
</tbody>
</table>
### Table 3-2  Setup Message List (contd.)

<table>
<thead>
<tr>
<th>Function</th>
<th>Header</th>
<th>Ref. page</th>
<th>Query</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Setup messages for file control</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deletion of files</td>
<td>File DElete Disk</td>
<td>3-66</td>
<td>Absent</td>
</tr>
<tr>
<td>Deletion of mass memory</td>
<td>File DElete Mass</td>
<td>3-67</td>
<td>Absent</td>
</tr>
<tr>
<td>Deletion of permanent memory</td>
<td>File DElete Permanent</td>
<td>3-68</td>
<td>Absent</td>
</tr>
<tr>
<td>Loading data from USB flash drive</td>
<td>File Load Disk data</td>
<td>3-72</td>
<td>Absent</td>
</tr>
<tr>
<td>Loading setup conditions from USB flash drive</td>
<td>File Load Disk Condition</td>
<td>3-73</td>
<td>Absent</td>
</tr>
<tr>
<td>Loading from mass memory</td>
<td>File Load Mass</td>
<td>3-74</td>
<td>Absent</td>
</tr>
<tr>
<td>Loading from permanent memory</td>
<td>File Load Permanent</td>
<td>3-75</td>
<td>Absent</td>
</tr>
<tr>
<td>Rename of file</td>
<td>File Rename Disk</td>
<td>3-76</td>
<td>Absent</td>
</tr>
<tr>
<td>Saving data to file</td>
<td>File Save Disk data</td>
<td>3-77</td>
<td>Absent</td>
</tr>
<tr>
<td>Saving setup conditions to file</td>
<td>File Save Disk Condition</td>
<td>3-78</td>
<td>Absent</td>
</tr>
<tr>
<td>Storage to mass memory</td>
<td>File Save Mass</td>
<td>3-79</td>
<td>Absent</td>
</tr>
<tr>
<td>Storage to permanent memory</td>
<td>File Save Permanent</td>
<td>3-80</td>
<td>Absent</td>
</tr>
<tr>
<td><strong>Setup messages for input part control</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Action for overload input</td>
<td>INput Action</td>
<td>3-81</td>
<td>Present</td>
</tr>
<tr>
<td>Arithmetic calculation for input (weighing)</td>
<td>INput CAcalculate</td>
<td>3-82</td>
<td>Present</td>
</tr>
<tr>
<td>Input equalization</td>
<td>INput Equalize</td>
<td>3-83</td>
<td>Present</td>
</tr>
<tr>
<td>Open equalizer setting</td>
<td>INput Open</td>
<td>3-84</td>
<td>Present</td>
</tr>
<tr>
<td>Detection level for overload input</td>
<td>INput OVer</td>
<td>3-85</td>
<td>Present</td>
</tr>
<tr>
<td>Short function setting</td>
<td>INput Short</td>
<td>3-86</td>
<td>Present</td>
</tr>
<tr>
<td><strong>Setup messages for measurement control</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Max cycles of automatic integration</td>
<td>Measure Auto MAx Cycle</td>
<td>3-87</td>
<td>Present</td>
</tr>
<tr>
<td>Max time of automatic integration</td>
<td>Measure Auto MAx Sec</td>
<td>3-88</td>
<td>Present</td>
</tr>
<tr>
<td>Type of max automatic integration</td>
<td>Measure Auto MAx Type</td>
<td>3-89</td>
<td>Present</td>
</tr>
<tr>
<td>Action of automatic integration</td>
<td>Measure Auto mode</td>
<td>3-90</td>
<td>Present</td>
</tr>
<tr>
<td>Coherence calculation mode</td>
<td>Measure Coherence</td>
<td>3-91</td>
<td>Present</td>
</tr>
<tr>
<td>Delay cycles for measurement start</td>
<td>Measure Delay Cycle</td>
<td>3-92</td>
<td>Present</td>
</tr>
<tr>
<td>Delay time for measurement start</td>
<td>Measure Delay Sec</td>
<td>3-93</td>
<td>Present</td>
</tr>
<tr>
<td>Delay type for measurement start</td>
<td>Measure Delay Type</td>
<td>3-94</td>
<td>Present</td>
</tr>
<tr>
<td>Order of harmonic analysis</td>
<td>Measure Harmonic</td>
<td>3-95</td>
<td>Present</td>
</tr>
<tr>
<td>Cycles of manual integration</td>
<td>Measure Integration Cycle</td>
<td>3-96</td>
<td>Present</td>
</tr>
<tr>
<td>Time of manual integration</td>
<td>Measure Integration Sec</td>
<td>3-97</td>
<td>Present</td>
</tr>
<tr>
<td>Type of manual integration</td>
<td>Measure Integration Type</td>
<td>3-98</td>
<td>Present</td>
</tr>
<tr>
<td>Measurement mode</td>
<td>Measure Mode</td>
<td>3-99</td>
<td>Present</td>
</tr>
<tr>
<td>Setup of Repeat Measure mode</td>
<td>Measure Repeat</td>
<td>3-100</td>
<td>Present</td>
</tr>
</tbody>
</table>
### Table 3-2 Setup Message List (contd.)

<table>
<thead>
<tr>
<th>Function</th>
<th>Header</th>
<th>Ref. page</th>
<th>Query</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Setup messages for oscillator control</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Output amplitude</td>
<td>OSCillator Amplitude</td>
<td>3-101</td>
<td>Present</td>
</tr>
<tr>
<td>Frequency</td>
<td>OSCillator Frequency</td>
<td>3-102</td>
<td>Present</td>
</tr>
<tr>
<td>Voltage change mode</td>
<td>OSCillator mode</td>
<td>3-103</td>
<td>Present</td>
</tr>
<tr>
<td>Output offset (DC bias)</td>
<td>OSCillator OFFSet</td>
<td>3-105</td>
<td>Present</td>
</tr>
<tr>
<td>Start/stop phase</td>
<td>OSCillator Phase</td>
<td>3-106</td>
<td>Present</td>
</tr>
<tr>
<td>Output waveform</td>
<td>OSCillator Waveform</td>
<td>3-107</td>
<td>Present</td>
</tr>
<tr>
<td><strong>Setup messages of various types</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Buzzer ON/OFF</td>
<td>SE tup Buzzer</td>
<td>3-108</td>
<td>Present</td>
</tr>
<tr>
<td>Execution of calibration</td>
<td>SE tup Calibration</td>
<td>3-109</td>
<td>Absent</td>
</tr>
<tr>
<td>Date of clock</td>
<td>SE tup Date</td>
<td>3-110</td>
<td>Present</td>
</tr>
<tr>
<td>GPIB header ON/OFF</td>
<td>SE tup Header</td>
<td>3-111</td>
<td>Present</td>
</tr>
<tr>
<td>Setup of initialization</td>
<td>SE tup Initialize</td>
<td>3-112</td>
<td>Absent</td>
</tr>
<tr>
<td>Mnemonic ON/OFF</td>
<td>SE tup Mnemonic</td>
<td>3-113</td>
<td>Present</td>
</tr>
<tr>
<td>Clock time</td>
<td>SE tup Time</td>
<td>3-114</td>
<td>Present</td>
</tr>
<tr>
<td><strong>Setup messages for sweep control</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Manual sweep</td>
<td>SW eep MA nual</td>
<td>3-115</td>
<td>Present</td>
</tr>
<tr>
<td>Start/stop of sweep measurement</td>
<td>SW eep ME a sure</td>
<td>3-116</td>
<td>Present</td>
</tr>
<tr>
<td>Sweep range</td>
<td>SW eep range</td>
<td>3-117</td>
<td>Present</td>
</tr>
<tr>
<td>Sweep resolution (Log steps/sweep)</td>
<td>SW eep RESolution log sweep</td>
<td>3-118</td>
<td>Present</td>
</tr>
<tr>
<td>Sweep resolution (Log steps/decade)</td>
<td>SW eep RESolution log Decade</td>
<td>3-118</td>
<td>Present</td>
</tr>
<tr>
<td>Sweep resolution (Lin steps/sweep)</td>
<td>SW eep RESolution Lin sweep</td>
<td>3-120</td>
<td>Present</td>
</tr>
<tr>
<td>Sweep resolution (Hz)</td>
<td>SW eep RESolution Lin Hz</td>
<td>3-121</td>
<td>Present</td>
</tr>
<tr>
<td>Type of sweep resolution</td>
<td>SW eep RESolution Mode</td>
<td>3-122</td>
<td>Present</td>
</tr>
<tr>
<td>Operation mode of low-speed &amp; high-density sweep</td>
<td>SW eep Slow mode</td>
<td>3-124</td>
<td>Present</td>
</tr>
<tr>
<td>Target channel for low-speed &amp; high-density sweep</td>
<td>SW eep Slow Target</td>
<td>3-125</td>
<td>Present</td>
</tr>
<tr>
<td>Target gain variation for low-speed &amp; high-density sweep (dB)</td>
<td>SW eep Slow Variation Logr</td>
<td>3-126</td>
<td>Present</td>
</tr>
<tr>
<td>Target amplitude variation for low-speed &amp; high-density sweep (Vms)</td>
<td>SW eep Slow Variation R</td>
<td>3-127</td>
<td>Present</td>
</tr>
<tr>
<td>Target phase variation for low-speed &amp; high-density sweep (deg)</td>
<td>SW eep Slow Variation Theta</td>
<td>3-128</td>
<td>Present</td>
</tr>
<tr>
<td>Target variation for real part of low-speed &amp; high-density sweep (Vms)</td>
<td>SW eep Slow Variation A</td>
<td>3-129</td>
<td>Present</td>
</tr>
<tr>
<td>Target variation for imaginary part of low-speed &amp; high-density sweep (Vrms)</td>
<td>SW eep Slow Variation B</td>
<td>3-130</td>
<td>Present</td>
</tr>
<tr>
<td>Target variation type for low-speed &amp; high-density sweep</td>
<td>SW eep Slow Variation Mode</td>
<td>3-131</td>
<td>Present</td>
</tr>
<tr>
<td><strong>List of other setup messages</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enabling SRQ transmission</td>
<td>SR qenable</td>
<td>3-135</td>
<td>Present</td>
</tr>
</tbody>
</table>
### 3.3.2 Query Message List

Table 3-3 Query Message List (contd.)

<table>
<thead>
<tr>
<th>Function</th>
<th>Header</th>
<th>Ref. Page</th>
<th>Setup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query messages for arithmetic calculations on data</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arithmetic calculation</td>
<td>?CA</td>
<td>lculation Arithmetic</td>
<td>3-20</td>
</tr>
<tr>
<td>Differentiation/integration</td>
<td>?CA</td>
<td>lculation Jw</td>
<td>3-22</td>
</tr>
<tr>
<td>Open-to-close loop conversion</td>
<td>?CA</td>
<td>lculation Loop</td>
<td>3-24</td>
</tr>
<tr>
<td>Query messages for amplitude compression control</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reference amplitude level</td>
<td>?CO</td>
<td>mpression Amplitude</td>
<td>3-26</td>
</tr>
<tr>
<td>Amplitude correction factor</td>
<td>?CO</td>
<td>mpression Correction</td>
<td>3-27</td>
</tr>
<tr>
<td>Max error rate of reference amplitude</td>
<td>?CO</td>
<td>mpression Error</td>
<td>3-28</td>
</tr>
<tr>
<td>Output amplitude limitation</td>
<td>?CO</td>
<td>mpression Limit</td>
<td>3-29</td>
</tr>
<tr>
<td>Amplitude compression mode</td>
<td>?CO</td>
<td>mpression mode</td>
<td>3-30</td>
</tr>
<tr>
<td>Max retry times of correction</td>
<td>?CO</td>
<td>mpression Retry</td>
<td>3-31</td>
</tr>
<tr>
<td>Query messages for data control</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tag-number switching</td>
<td>?DA</td>
<td>ta CUrrent</td>
<td>3-32</td>
</tr>
<tr>
<td>Display data switching</td>
<td>?DA</td>
<td>ta Display</td>
<td>3-33</td>
</tr>
<tr>
<td>Transfer of setup conditions to controller</td>
<td>?DA</td>
<td>ta Read COndition</td>
<td>3-36</td>
</tr>
<tr>
<td>Data transfer to controller</td>
<td>?DA</td>
<td>ta Read CUrrent</td>
<td>3-39</td>
</tr>
<tr>
<td>Transfer of tag data to controller</td>
<td>?DA</td>
<td>ta Read data</td>
<td>3-40</td>
</tr>
<tr>
<td>Block count of tag data</td>
<td>?DA</td>
<td>ta Read S</td>
<td>ize</td>
</tr>
<tr>
<td>Title of tag data</td>
<td>?DA</td>
<td>ta Read T</td>
<td>itle</td>
</tr>
<tr>
<td>Data transfer format</td>
<td>?DA</td>
<td>ta Template</td>
<td>3-44</td>
</tr>
<tr>
<td>Query messages for display control</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Analysis mode</td>
<td>?DI</td>
<td>isplay ANalysis</td>
<td>3-48</td>
</tr>
<tr>
<td>Autoscale</td>
<td>?DI</td>
<td>isplay AUto</td>
<td>3-49</td>
</tr>
<tr>
<td>Grid ON/OFF</td>
<td>?DI</td>
<td>isplay Grid mode</td>
<td>3-50</td>
</tr>
<tr>
<td>Grid type</td>
<td>?DI</td>
<td>isplay Grid T</td>
<td>ype</td>
</tr>
<tr>
<td>Grid mode</td>
<td>?DI</td>
<td>isplay Grid S</td>
<td>yle</td>
</tr>
<tr>
<td>Setup of active marker</td>
<td>?DI</td>
<td>isplay MA</td>
<td>rker Active</td>
</tr>
<tr>
<td>Setup of delta marker mode</td>
<td>?DI</td>
<td>isplay MA</td>
<td>rker mode</td>
</tr>
<tr>
<td>Graph display mode</td>
<td>?DI</td>
<td>isplay mode</td>
<td>3-56</td>
</tr>
<tr>
<td>Phase display range</td>
<td>?DI</td>
<td>isplay Phase</td>
<td>3-60</td>
</tr>
<tr>
<td>Display scale (X axis)</td>
<td>?DI</td>
<td>isplay Scale Xaxis</td>
<td>3-61</td>
</tr>
<tr>
<td>Display scale (Y\textsubscript{1} axis)</td>
<td>?DI</td>
<td>isplay Scale Y1axis</td>
<td>3-62</td>
</tr>
<tr>
<td>Display scale (Y\textsubscript{2} axis)</td>
<td>?DI</td>
<td>isplay Scale Y2axis</td>
<td>3-63</td>
</tr>
<tr>
<td>Setup of Display Unit</td>
<td>?DI</td>
<td>isplay Units</td>
<td>3-64</td>
</tr>
<tr>
<td>Graph display window mode</td>
<td>?DI</td>
<td>isplay W</td>
<td>indow</td>
</tr>
</tbody>
</table>
### Table 3-3 Query Message List (contd.)

<table>
<thead>
<tr>
<th>Function</th>
<th>Header</th>
<th>Ref. page</th>
<th>Setup</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Query messages for file control</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>List of files</td>
<td>?File DIr Disk</td>
<td>3-69</td>
<td>Absent</td>
</tr>
<tr>
<td>Mass memory list</td>
<td>?File DIr Mass</td>
<td>3-70</td>
<td>Absent</td>
</tr>
<tr>
<td>Permanent memory list</td>
<td>?File DIr Permanent</td>
<td>3-71</td>
<td>Absent</td>
</tr>
<tr>
<td><strong>Setup messages for input part control</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Action for overload input</td>
<td>?INput Action</td>
<td>3-81</td>
<td>Present</td>
</tr>
<tr>
<td>Arithmetic calculation for input</td>
<td>?INput CAalculate</td>
<td>3-82</td>
<td>Present</td>
</tr>
<tr>
<td>(weighing)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Input equalization</td>
<td>?INput Equalize</td>
<td>3-83</td>
<td>Present</td>
</tr>
<tr>
<td>Open equalization</td>
<td>?INput Open</td>
<td>3-84</td>
<td>Present</td>
</tr>
<tr>
<td>Detection level for overload input</td>
<td>?INput Over</td>
<td>3-85</td>
<td>Present</td>
</tr>
<tr>
<td>Short equalization</td>
<td>?INput Short</td>
<td>3-86</td>
<td>Present</td>
</tr>
<tr>
<td><strong>Setup messages for measurement control</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Max cycles of automatic integration</td>
<td>?Measure Auto MAx Cycle</td>
<td>3-87</td>
<td>Present</td>
</tr>
<tr>
<td>Max time of automatic integration</td>
<td>?Measure Auto MAx Sec</td>
<td>3-88</td>
<td>Present</td>
</tr>
<tr>
<td>Type of max automatic integration</td>
<td>?Measure Auto MAx Type</td>
<td>3-89</td>
<td>Present</td>
</tr>
<tr>
<td>Action of automatic integration</td>
<td>?Measure Auto mode</td>
<td>3-90</td>
<td>Present</td>
</tr>
<tr>
<td>Coherence calculation mode</td>
<td>?Measure Coherence</td>
<td>3-91</td>
<td>Present</td>
</tr>
<tr>
<td>Delay cycles for measurement start</td>
<td>?Measure Delay Cycle</td>
<td>3-92</td>
<td>Present</td>
</tr>
<tr>
<td>Delay time for measurement start</td>
<td>?Measure Delay Sec</td>
<td>3-93</td>
<td>Present</td>
</tr>
<tr>
<td>Delay type for measurement start</td>
<td>?Measure Delay Type</td>
<td>3-94</td>
<td>Present</td>
</tr>
<tr>
<td>Order of harmonic analysis</td>
<td>?Measure Harmonic</td>
<td>3-95</td>
<td>Present</td>
</tr>
<tr>
<td>Cycles of manual integration</td>
<td>?Measure Integration Cycle</td>
<td>3-96</td>
<td>Present</td>
</tr>
<tr>
<td>Time of manual integration</td>
<td>?Measure Integration Sec</td>
<td>3-97</td>
<td>Present</td>
</tr>
<tr>
<td>Type of manual integration</td>
<td>?Measure Integration Type</td>
<td>3-98</td>
<td>Present</td>
</tr>
<tr>
<td>Measurement mode</td>
<td>?Measure Mode</td>
<td>3-99</td>
<td>Present</td>
</tr>
<tr>
<td>Setup of Repeat Measure mode</td>
<td>?Measure Repeat</td>
<td>3-100</td>
<td>Present</td>
</tr>
<tr>
<td><strong>Query messages for oscillator control</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Output amplitude</td>
<td>?OScillator Amplitude</td>
<td>3-101</td>
<td>Present</td>
</tr>
<tr>
<td>Frequency</td>
<td>?OScillator Frequency</td>
<td>3-102</td>
<td>Present</td>
</tr>
<tr>
<td>Voltage reset mode</td>
<td>?OScillator mode</td>
<td>3-103</td>
<td>Present</td>
</tr>
<tr>
<td>Output offset (DC bias)</td>
<td>?OScillator OFFSset</td>
<td>3-105</td>
<td>Present</td>
</tr>
<tr>
<td>Start/stop phase</td>
<td>?OScillator Phase</td>
<td>3-106</td>
<td>Present</td>
</tr>
<tr>
<td>Output waveform</td>
<td>?OScillator Waveform</td>
<td>3-107</td>
<td>Present</td>
</tr>
</tbody>
</table>
### Table 3-3 Query Message List (contd.)

<table>
<thead>
<tr>
<th>Function</th>
<th>Header</th>
<th>Ref. page</th>
<th>Setup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query messages of various types</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Buzzer ON/OFF</td>
<td>?SEtp Buzzer</td>
<td>3-108</td>
<td>Present</td>
</tr>
<tr>
<td>Date of clock</td>
<td>?SEtp Date</td>
<td>3-110</td>
<td>Present</td>
</tr>
<tr>
<td>GPIB header ON/OFF</td>
<td>?SEtp Header</td>
<td>3-111</td>
<td>Present</td>
</tr>
<tr>
<td>Mnemonic ON/OFF</td>
<td>?SEtp Mnemonic</td>
<td>3-113</td>
<td>Present</td>
</tr>
<tr>
<td>Clock time</td>
<td>?SEtp Time</td>
<td>3-114</td>
<td>Present</td>
</tr>
<tr>
<td>Query messages for sweep control</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Manual sweep</td>
<td>?SWep MA nual</td>
<td>3-115</td>
<td>Present</td>
</tr>
<tr>
<td>Start/stop of sweep measurement</td>
<td>?SWep ME asure</td>
<td>3-116</td>
<td>Present</td>
</tr>
<tr>
<td>Sweep range</td>
<td>?SWep range</td>
<td>3-117</td>
<td>Present</td>
</tr>
<tr>
<td>Sweep resolution (Log steps/sweep)</td>
<td>?SWep RE solution log sweep</td>
<td>3-118</td>
<td>Present</td>
</tr>
<tr>
<td>Sweep resolution (Log steps/decade)</td>
<td>?SWep RE solution log Decade</td>
<td>3-118</td>
<td>Present</td>
</tr>
<tr>
<td>Sweep resolution (Lin steps/sweep)</td>
<td>?SWep RE solution LIn sweep</td>
<td>3-120</td>
<td>Present</td>
</tr>
<tr>
<td>Sweep resolution (Hz)</td>
<td>?SWep RE solution LIn HZ</td>
<td>3-121</td>
<td>Present</td>
</tr>
<tr>
<td>Type of sweep resolution</td>
<td>?SWep RE solution Mode</td>
<td>3-122</td>
<td>Present</td>
</tr>
<tr>
<td>Operation mode of low-speed &amp; high-density sweep</td>
<td>?SWep Slow mode</td>
<td>3-124</td>
<td>Present</td>
</tr>
<tr>
<td>Target channel for low-speed &amp; high-density sweep</td>
<td>?SWep Slow Target</td>
<td>3-125</td>
<td>Present</td>
</tr>
<tr>
<td>Target gain variation for low-speed &amp; high-density sweep (dB)</td>
<td>?SWep Slow Variation Logr</td>
<td>3-126</td>
<td>Present</td>
</tr>
<tr>
<td>Target amplitude variation for low-speed &amp; high-density sweep (Vms)</td>
<td>?SWep Slow Variation R</td>
<td>3-127</td>
<td>Present</td>
</tr>
<tr>
<td>Target phase variation for low-speed &amp; high-density sweep (deg)</td>
<td>?SWep Slow Variation Theta</td>
<td>3-128</td>
<td>Present</td>
</tr>
<tr>
<td>Target variation for real part of low-speed &amp; high-density sweep (Vms)</td>
<td>?SWep Slow Variation A</td>
<td>3-129</td>
<td>Present</td>
</tr>
<tr>
<td>Target variation for imaginary part of low-speed &amp; high-density sweep (Vms)</td>
<td>?SWep Slow Variation B</td>
<td>3-130</td>
<td>Present</td>
</tr>
<tr>
<td>Target variation type for low-speed &amp; high-density sweep</td>
<td>?SWep Slow Variation Mode</td>
<td>3-131</td>
<td>Present</td>
</tr>
<tr>
<td>List of other query messages</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Error code</td>
<td>?Error</td>
<td>3-132</td>
<td>Absent</td>
</tr>
<tr>
<td>Model identification code</td>
<td>?IDentifier</td>
<td>3-133</td>
<td>Absent</td>
</tr>
<tr>
<td>Results of overload detection</td>
<td>?OVERload</td>
<td>3-134</td>
<td>Absent</td>
</tr>
<tr>
<td>Enabling SRQ transmission</td>
<td>?SRQenable</td>
<td>3-135</td>
<td>Present</td>
</tr>
<tr>
<td>Status byte</td>
<td>?Status</td>
<td>3-136</td>
<td>Absent</td>
</tr>
<tr>
<td>Software version</td>
<td>?Version</td>
<td>3-137</td>
<td>Absent</td>
</tr>
<tr>
<td>Reading of all setup conditions</td>
<td>?Learn</td>
<td>3-138</td>
<td>Absent</td>
</tr>
</tbody>
</table>
3.4 Individual program code description

In paragraph 3.4, the program codes of the GPIB/USB will be detailed.

The lowercase part of each program code can be omitted.

The underlined part of an answer to a query message is a header that is output only when "SEtup Header" is set to 1 or ON.
### CAIculation Arithmetic

**?CAIculation Arithmetic**

**Setup:** 
CAIculation Arithmetic <param1>, <param2>, <param3>, <param4>, 
<param5>, <param6>

**Query:** 
?CAIculation Arithmetic  
(Answer)  
CALCULATION ARITHMETIC <param1>, <param2>, <param3>, 
<param4>, <param5>, <param6>

- **General:** Sets or inquires the function of arithmetic calculation

- **Parameter (when parameters are set)**
  - **param1:** Type of target data 1 for arithmetic calculation
  - **param4:** Type of target data 2 for arithmetic calculation

**Format:** NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Tag</td>
<td>Tag data</td>
</tr>
<tr>
<td>1</td>
<td>Constant</td>
<td>Real constant</td>
</tr>
<tr>
<td>2</td>
<td>J</td>
<td>Imaginary constant</td>
</tr>
<tr>
<td></td>
<td>Omit</td>
<td>None changes</td>
</tr>
</tbody>
</table>

- **param2:** Value of target data 1 for arithmetic calculation
- **param5:** Value of target data 2 for arithmetic calculation

**Format:** NR1 (when target data type for calculation is tag data)

- **Range:** 0, 1 to 6 (0 indicates the tag currently displayed)

**NR3 (when target data type for calculation is a (real or imaginary) constant)**

- **Range:** -99.99E+6 to +99.99E+6
- **Resolution:** 0.001E-06
- **Omit:** None changes

- **param3:** Arithmetic calculation mode

**Format:** NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Add</td>
<td>Addition  (param2) + (param5) = (param6)</td>
</tr>
<tr>
<td>1</td>
<td>Subtract</td>
<td>Subtraction  (param2) – (param5) = (param6)</td>
</tr>
<tr>
<td>2</td>
<td>Multiply</td>
<td>Multiplication  (param2) x (param5) = (param6)</td>
</tr>
<tr>
<td>3</td>
<td>Divide</td>
<td>Division  (param2) ÷ (param5) = (param6)</td>
</tr>
<tr>
<td></td>
<td>Omit</td>
<td>None changes</td>
</tr>
</tbody>
</table>
3.4 Individual program code description

- **param6:** Tag number to which calculation result is stored
  - **Format:** NR1
  - **Range:** 0, 1 to 6 (0 indicates the tag currently displayed)
  - **Omit:** None changes

- **Answer message (when a query is run)**
  - **param1:** Current type of target data 1 for arithmetic calculation
  - **param4:** Current type of target data 2 for arithmetic calculation
  - **Format:** NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NR1 String</td>
<td></td>
</tr>
<tr>
<td>0 TAG</td>
<td>Target data is tags</td>
</tr>
<tr>
<td>1 CONSTANT</td>
<td>Target data is real constants</td>
</tr>
<tr>
<td>2 J</td>
<td>Target data is imaginary constants</td>
</tr>
</tbody>
</table>

- **param2:** Current value of target data 1 for arithmetic calculation
- **param5:** Current value of target data 2 for arithmetic calculation
  - **Format:** NR1 (when target data type for calculation is tag data)
    - **Range:** 0, 1 to 6 (0 indicates the tag currently displayed)
    - **Characters:** 2
  - **NR3** (when target data type for calculation is a (real or imaginary) constant)
    - **Range:** -99.99E+6 to +99.99E+6
    - **Characters:** 10

- **param3:** Arithmetic calculation mode
  - **Format:** NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NR1 String</td>
<td></td>
</tr>
<tr>
<td>0 ADD</td>
<td>Addition</td>
</tr>
<tr>
<td>1 SUBSTRACT</td>
<td>Subtraction</td>
</tr>
<tr>
<td>2 MULTIPLY</td>
<td>Multiplication</td>
</tr>
<tr>
<td>3 DIVIDE</td>
<td>Division</td>
</tr>
</tbody>
</table>

- **param6:** Tag number to which calculation result is stored
  - **Format:** NR1
  - **Range:** 0, 1 to 6 (0 indicates the tag currently displayed)
  - **Characters:** 2

- **Example setup**
  - `cal a 0,0,2,1,10,0` Multiplies the data currently displayed by 10
3.4 Individual program code description

**CAlculation Jw**

?CAlculation Jw

Setup: `CAlculation Jw <param1>, <param2>, <param3>`
Query: `?CAlculation Jw`

(Answer) `CALCULATION JW <param1>, <param2>, <param3>`

- General: Sets or inquires the function of differential and integral operation

- Parameter (when parameters are set)
  - param1: Type of arithmetic calculation
    Format: NR1
    | param1 | Description          |
    |--------|----------------------|
    | -2     | Double integration   |
    | -1     | Integration          |
    | 1      | Differentiation      |
    | 2      | Second order         |
    |        | differentiation       |
    | (Omit) | None changes         |
  - param2: Tag number of target data for arithmetic calculation
    Format: NR1
    Range: 0, 1 to 6 (0 indicates the tag currently displayed)
    Omit: None changes
  - param3: Tag number to which calculation result is stored
    Format: NR1
    Range: 0, 1 to 6 (0 indicates the tag currently displayed)
    Omit: None changes
3.4 Individual program code description

- Answer message (when a query is run)
  - param1: Current type of arithmetic calculation
    Format: NR1
    | param1 | Description         |
    |--------|--------------------|
    | -2     | Double integration |
    | -1     | Integration         |
    | 1      | Differentiation     |
    | 2      | Second order        |
    |        | differentiation      |

- param2: Current tag number of target data for arithmetic calculation
  Format: NR1
  Range: 0, 1 to 6 (0 indicates the tag currently displayed)
  Characters: 2

- param3: Tag number to which current calculation result is stored
  Format: NR1
  Range: 0, 1 to 6 (0 indicates the tag currently displayed)
  Characters: 2

- Example setup
  cal jw -2,1,2
  Double-integrates tag 1 data and store the result to tag 2
3.4 Individual program code description

**CALculation Loop**

**?CALculation Loop**

Setup:  **CALculation Loop** <param1>, <param2>, <param3>, <param4>, <param5>
Query:  **CALculation Loop**

(Answer)  **CALCULATION LOOP** <param1>, <param2>, <param3>, <param4>, <param5>

- General: Sets or inquires the function of open-to-close loop calculation

- Parameter (when parameters are set)
  - **param1:**  Tag number of target data for arithmetic calculation
    Format:  NR1
    Range:  0, 1 to 6 (0 indicates the tag currently displayed)
    Omit:  None changes
  - **param2:**  Type of feedback element data Tm
    Format:  NR1 or string
    | NR1 | String     | Description       |
    |-----|------------|-------------------|
    | 0   | Tag        | Tag data          |
    | 1   | Constant   | Real constant     |
    |     | Omit       | None changes      |
  - **param3:**  Value of feedback element data Tm
    Format:  NR1 (when target data type for calculation is tag data)
    Range:  0, 1 to 6 (0 indicates the tag currently displayed)
    NR3 (when target data type for calculation is a constant)
    Range:  -099.99E+6 to +99.99E+6
    Resolution:  0.001E-6
  - **param4:**  Arithmetic calculation mode
    Format:  NR1 or string
    | NR1 | String     | Description       |
    |-----|------------|-------------------|
    | 0   | Close      | Open→close loop conversion |
    | 1   | Open       | Close→open loop conversion |
    |     | Omit       | None changes      |
  - **param5:**  Tag number to which calculation result is stored
    Format:  NR1
    Range:  0, 1 to 60 (0 indicates the tag currently displayed)
• **Answer message (when a query is run)**
  - **param1:** Current tag number of target data for arithmetic calculation
    Format: NR1
    Range: 0, 1 to 6 (0 indicates the tag currently displayed)
    Characters: 2
  - **param2:** Current type of feedback element data Tm
    Format: NR1 (characters: 2) or string (to be set by S\textit{E}tup \textit{M}nemonic)
    | Answer format | Description          |
    |----------------|----------------------|
    | 0 TAG          | Target data is tags  |
    | 1 CONSTANT     | Target data is real constants |
  - **param3:** Current value of feedback element data Tm
    Format: NR1 (when feedback element data is tag data)
    Range: 0, 1 to 6 (0 indicates the tag currently displayed)
    Characters: 2
    NR3 (when feedback element data is real constants)
    Characters: 10
  - **param4:** Current mode of arithmetic calculation
    Format: NR1 (characters: 2) or string (to be set by S\textit{E}tup \textit{M}nemonic)
    | Answer format | Description          |
    |----------------|----------------------|
    | 0 CLOSE        | Open$\rightarrow$close loop conversion |
    | 1 OPEN         | Close$\rightarrow$open loop conversion |
  - **param5:** Tag number to which current calculation result is stored
    Format: NR1
    Range: 0, 1 to 6 (0 indicates the tag currently displayed)
    Characters: 2

• Example setup
  \texttt{cal loop 1, 0, 2, 1, 3}
  Determines the open loop property and stores it to tag 3 when the property data of close loop is in tag 1 and the property data of feedback element Tm is in tag 2
3.4 Individual program code description

**COmpression Amplitude**

**?COmpression Amplitude**

Setup:  COmpression Amplitude <param>
Query:  ?COmpression Amplitude
        (Answer)  COMPRESSION AMPLITUDE <param>

- **General:** Sets or inquires the reference amplitude level for amplitude compression

- **Parameter (when parameters are set)**
  - param:  Reference amplitude level
    Format:  NR3
    Range:  1E-3 to 250(1 mVrms to 250 Vrms)
    Resolution:  3 digits

- **Answer message (when a query is run)**
  - param:  Current level of reference amplitude
    Format:  NR3
    Range:  1.00E-03 to 250E+00(1 mVrms to 250 Vrms)
    Characters:  9

- **Example setup**
  compression amplitude 7.07e1  Reference amplitude level: 70.7 Vrms

- **Related program codes**
  COmpression Correction, COmpression Error, COmpression Limit,
  COmpression mode, COmpression Retry
3.4 Individual program code description

- **COmpression Correction**

  **?COmpression Correction**

  **Setup:** `COMpression Correction <param>`
  **Query:** `?COMpression Correction`
  **(Answer)** `COMPRESSION CORRECTION <param>`

  - **General:** Sets or inquires the amplitude correction factor for amplitude compression

  - **Parameter (when parameters are set)**
    - `param:` Amplitude correction factor
      - **Format:** NR1
      - **Range:** 0 to 100 (%)
      - **Resolution:** 1 (%)

  - **Answer message (when a query is run)**
    - `param:` Current amplitude correction factor
      - **Format:** NR1
      - **Range:** 0 to 100 (%)
      - **Characters:** 4

  - **Example setup**
    
    compression correction 70  
    Amplitude correction factor: 70%

  - **Related program codes**
    - COMpression Amplitude, COMpression Error, COMpression Limit,
    - COMpression mode, COMpression Retry
### COMpression Error

**?COMpression Error**

Setup:  `COMpression Error <param>`
Query:  `?COMpression Error`  
  (Answer)  `COMPRESSION ERROR <param>`

- General: Sets or inquires the maximum error rate for amplitude compression

- Parameter (when parameters are set)
  - `param`: Maximum error rate
    - Format: NR1
    - Range: 0 to 100 (%)
    - Resolution: 1 (%)

- Answer message (when a query is run)
  - `param`: Current maximum error rate
    - Format: NR1
    - Range: 0 to 100 (%)
    - Characters: 4

- Example setup
  
  compression error 1  Maximum error rate: 1%

- Related program codes
  
  COMpression Amplitude, COMpression Correction, COMpression Limit,
  COMpression mode, COMpression Retry
### COMpression Limit

**?COMpression Limit**

**Setup:** COMpression Limit <param>

**Query:** ?COMpression Limit

(Answer) COMPRESSION LIMIT <param>

- **General:** Sets or inquires the output amplitude limitation for amplitude compression

- **Parameter (when parameters are set)**
  - **param:** Maximum amplitude (output open conversion)
    - Format: NR3
    - Range: 1E-3 to 10.0 (Vpeak)
    - Resolution: Up to 3 digits, 10 μV

- **Answer message (when a query is run)**
  - **param:** Current maximum amplitude (output open conversion)
    - Format: NR3
    - Range: 1.00E-03 to 10.0E+00 (Vpeak)
    - Characters: 9

- **Example setup**
  
  compression limit 1.2  
  
  Maximum output amplitude: 1.2 Vpeak

- **Related program codes**
  - COMpression Amplitude, COMpression Correction, COMpression Error,
  - COMpression mode, COMpression Retry
### COMpression mode

**Setup:**  **COMpression mode** <param>

**Query:**  **?COMpression mode**

(Answer)  **COMPRESSION MODE** <param>

- General: Sets or inquires the operation mode of amplitude compression

- Parameter (when parameters are set)
  - param: Operation mode of amplitude compression
    
    Format: NR1 or string

    | NR1 | String | Description |
    |-----|--------|-------------|
    | 0   | OFF   | Amplitude compression OFF |
    | 1   | CH1   | Amplitude compression ON, Reference input: CH1 |
    | 2   | CH2   | Amplitude compression ON, Reference input: CH2 |
    |     | Omit  | None changes |

- Answer message (when a query is run)
  - param: Current operation mode of amplitude compression
    
    Format: NR1 (characters: 2) or string (to be set by **SE**tup **M**nemonic)

    | Answer format | Description |
    |---------------|-------------|
    | NR1           |             |
    | 0             | OFF Amplitude compression OFF |
    | 1             | CH1 Amplitude compression ON, Reference input: CH1 |
    | 2             | CH2 Amplitude compression ON, Reference input: CH2 |

- Example setup
  
  compression mode ch1  
  Amplitude compression ON with ch1 as the reference input

- Related program codes
  
  COMpression Amplitude, COMpression Correction, COMpression Error, COMpression Limit, COMpression Retry
3.4 Individual program code description

- **COmpression Retry**

  ?COmpression Retry

  Setup:  COmpression Retry <param>

  Query: ?COmpression Retry

  (Answer)  COMPRESSION RETRY <param>

  - **General:** Sets or inquires the maximum retry times of correction for amplitude compression

  - **Parameter (when parameters are set)**
    - **param:** Maximum retry times of correction
      - Format: NR1
      - Range: 1 to 9999 (times)
      - Resolution: 1

  - **Answer message (when a query is run)**
    - **param:** Current maximum retry times of correction
      - Format: NR1
      - Range: 1 to 9999 (times)
      - Characters: 5

  - **Example setup**
    - compression retry 100
    - Maximum retry times of correction: 100

  - **Related program codes**
    - COMpression Amplitude, COMpression Correction, COMpression Error,
    - COMpression Limit, COMpression mode
3.4 Individual program code description

- **DAta CUrrent**

  **?DAta CUrrent**

  Setup:   **DAta CUrrent <param>**
  Query:  **?DAta CUrrent**
  (Answer)  **DATA CURRENT <param>**

  • General:  Sets or inquires the current tag number

  • Parameter (when parameters are set)
    • param:  Tag number to be defined as the current tag number
      Format:  NR1
      Range:  1 to 6

  • Answer message (when a query is run)
    • param:  Current current tag number
      Format:  NR1
      Range:  1 to 6
      Characters:  2

  • Example setup
    data current 2              Defines and displays tag 2 as the current tag number

  • Remarks
    The current tag indicates the one that is currently active. It refers to the tag number that is used when tag number 0 is specified in arithmetic calculation, etc.
    The following will result if a tag is specified as a current tag when a graph is shown in SPLIT view mode ("DIsplay Window SPLIT"):  
      • When the specified tag number matches the tag number of either the upper or lower graph, the graph will become active.  
      • When the specified tag number is not currently displayed, the data of that number will be assigned to either the upper or lower graph which is currently active.

  • Related program codes
    DIsplay Window , DAta Display
3.4 Individual program code description

■ DAte Display

?DAte Display

Setup:  DAte Display <param1>, <param2>
Query:  ?DAte Display
       (Answer) DATA DISPLAY <param1>, <param2>

• General: Sets or inquires the tag number to be displayed

• Parameter (when parameters are set)
  • param1: Tag number to be displayed. In SPLIT view mode, this is the tag number that is displayed in the upper graph.
    Format: NR1
    Range: 1 to 6
    Omit: None changes
  • param2: In SPLIT view mode, this is the tag number that is displayed in the lower graph. This will be disabled in SINGLE view mode. If 0 is specified, then the lower graph will not be changed.
    Format: NR1
    Range: 0 to 6
    Omit: None changes

• Answer message (when a query is run)
  • param1: Tag number that is currently being displayed (in SPLIT view mode, the tag number in the upper graph)
    Format: NR1
    Range: 1 to 6
    Characters: 2
  • param2: Tag number that is displayed in the lower graph (in SPLIT view mode). Constantly 0 in SINGLE view mode
    Format: NR1
    Range: 1 to 6
    Characters: 2

• Example setup
  data display 1,2 Displays tag 1 and tag 2 respectively in the upper and lower graphs (in SPLIT mode)

• Related program codes
  DIisplay Window, DAte CUrrent
3.4 Individual program code description

**DA**ta **E**qualize

Setup:  **DA**ta **E**qualize <param>

- **General**: Registers with equalize memory

- **Parameter (when parameters are set)**
  - **param**: Tag number of data to be registered with equalize memory
    - **Format**: NR1
    - **Range**: 1 to 6

- **Example setup**
  - data equalize 1  Registers tag 1 data with equalize memory

- **Remarks**
  - After the data has been registered with equalize memory, the equalize function will be enabled by means of "INput Equalize ON"

- **Related program code**
  - INput Equalize
3.4 Individual program code description

■ DATA Open

Setup:  DATA Open <param>

- General:  Register in open equalizer memory

- Parameter (when setting)
  - param:  Tag number for registry in open equalizer memory
    Format:  NR1
    Range:  1 to 6

- Example setup
  data open 1  Register tag 1 data in open equalizer memory

- Remarks
  After registering, the open equalizer is effective by INput Open ON.

- Related program code
  INput Open
3.4 Individual program code description

**?DA?ta Read COndition**

Query:  **?DA?ta Read COndition <param>**

(Answer)

DATA READ CONDITION

<param1>, <param2>, <param3> (block delimiter)
<param4> (block delimiter)
<param5>, <param6>, <param7> (block delimiter)
<param8>, <param9>, <param10>, <param11> (block delimiter)
<param12>, <param13> (block delimiter)
<param14>, <param15> (block delimiter)
<param16>, <param17>, <param18>, <param19>, <param20> (record delimiter)

- General:  Inquires the measurement conditions for specified tag data

- Parameter (when a query is run)
  - param:  Tag number to which measurement conditions will be transferred
    Format:  NR1
    Range:  1 to 6
    Omit:  Current tag number

- Answer message
  - param1:  Data type
    Format:  NR1 (characters: 2) or string (to be set by **SE?tup Mnemonic**)
    | Answer format | Description     |
    |---------------|----------------|
    | NR1 String    |                |
    | 0 RAW         | Measurement data|
    | 1 OPRD        | Operated data   |

  - param2:  Data count
    Format:  NR1
    Range:  Up to (Data count of specified tag)
    Characters:  6

  - param3:  Date and time when a measurement was carried out
    Format:  YYYY-MM-DD, HH-MM-SS (year-month-day, hour-minute-second)
    SS is constantly 00
3.4 Individual program code description

- param4: Title of data  
  Format: A character string is enclosed with double quotation codes ("").  
  Refer to "3.1.3 b) Parameters in string format".
- param5: Oscillator amplitude ($V_{\text{peak}}$)  
  Format: NR3, 9 characters
- param6: Oscillator, DC bias (V)  
  Format: NR3, 10 characters
- param7: Oscillator waveform  
  Format: NR1 or string  
  Refer to "Oscillator Waveform".
- param8: Maximum sweep frequency (Hz)  
  Format: NR3, 17 characters
- param9: Minimum sweep frequency (Hz)  
  Format: NR3, 17 characters
- param10: Sweep type  
  Format: NR1 or string  
  Refer to "Sweep RESolution Mode".
- param11: Sweep resolution  
  Format: NR1 or string  
  Refer to "Sweep RESolution ***".
- param12: Integration type  
  Format: NR1 or string  
  Refer to "Measure Integration Type".
- param13: Integration level  
  Format: NR1 or NR3  
  Refer to "Measure Integration ***".
- param14: Delay type  
  Format: NR1 or string  
  Refer to "Measure Delay Type".
- param15: Delay level  
  Format: NR1 or NR3  
  Refer to "Measure Delay ***".
- param16: Degree of harmonic analysis  
  Format: NR1, 3 characters
- param17: Measurement mode  
  Format: NR1 or string  
  Refer to "Measure Mode".
- param18: Automatic integration function  
  Format: NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NR1</td>
<td>String</td>
</tr>
<tr>
<td>0</td>
<td>OFF</td>
</tr>
<tr>
<td>1</td>
<td>ON</td>
</tr>
</tbody>
</table>
3.4 Individual program code description

- **param19:** Low-speed & high-density sweep function
  
  Format: NR1 (characters: 2) or string (to be set by **SE**tup **M**nemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NR1</td>
<td>String</td>
</tr>
<tr>
<td>0</td>
<td>OFF Normal sweep</td>
</tr>
<tr>
<td>1</td>
<td>ON Low-speed &amp; high-density sweep</td>
</tr>
</tbody>
</table>

- **param20:** Amplitude compression function
  
  Format: NR1 (characters: 2) or string (to be set by **SE**tup **M**nemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NR1</td>
<td>String</td>
</tr>
<tr>
<td>0</td>
<td>OFF Amplitude compression OFF</td>
</tr>
<tr>
<td>1</td>
<td>ON Amplitude compression ON</td>
</tr>
</tbody>
</table>

- **Example setup**

  ?data read condition 1 Transfers the measurement conditions of tag data 1 to controller

- **Remarks**

  When the data is of operated type (param1 is 0 or OPRD), the data between param5 and param20 is invalid.
3.4 Individual program code description

- **?DATa Read CUrent**

  Query:   **?DATa Read CUrent**

  (Answer) Measurement data

  - General: Inquires the final data that was measured (block 1)

  - Answer message (when a query is run)
    The format that was set by "DATa Template" will be used.

  - Example setup
    ?data read current

  - Remarks
    The transfer format etc. can be set by "DATa Template". No header is included.

  - Related program code
    DATa Template
### ?DAta Read data

**Query:** ?DAta Read data <param1>, <param2>, <param3>  
(Answer) (Tag data)

- **General:** Inquires a specified data range of a specified tag

- **Parameter (when parameters are set)**
  - **param1:** Tag number to be inquired for data  
    - Format: NR1  
    - Range: 1 to 6  
    - Omit: Current tag number
  - **param2:** Block number of data from which reading is started  
    - Format: NR1  
    - Range: 0 to (block count of data to be read minus 1)  
    - Omit: 0
  - **param3:** Block count of data to be read  
    - Format: NR1  
    - Range: 1 to (block count of data to be read)  
    - Omit: Block count of data to be read

- **Answer message (when a query is run)**  
The format set by "DAta Template" will be used.

- **Example setup**
  - `?data read data 2` Inquires all data of tag 2

- **Remarks**  
The transfer format etc. can be set using "DAta Template" No header is included.  
An error will result if an attempt is made to read the data of a tag for which a measurement is in progress (error code:43).

- **Related program codes**  
  - DAta Template, ?DAta Read Size, DAta Write data
3.4 Individual program code description

■ ?DAta Read Size

Query:  ?DAta Read Size <param>
(Answer)  DATA READ SIZE <param>

- General: Inquiries the data block count of a tag

- Parameter (when a query is run)
  - param: Tag number from which the data block count is read out
    Format: NR1
    Range: 1 to 6
    Omit: Current tag number

- Answer message (when a query is run)
  - param: Block count of data in a specified tag
    Format: NR1
    Range: 1 to (block count of data in a specified tag)
    Characters: 6

- Example setup
  ?data read size 1  Inquires the block count of data in tag 1

- Related program code
  ?DAta Read data
3.4 Individual program code description

■ ?DAta Read Title

Query:  ?DAta Read Title <param>

(Answer)  DATA WRITE TITLE <param>

- General: Inquires the title string of a tag

- Parameter (when a query is run)
  - param: Tag number from which the title string is read out
    Format: NR1
    Range: 1 to 6
    Omit: Current tag number

- Answer message (when a query is run)
  - param: Title string of data in a specified tag
    Format: String

- Example setup
  ?data read title 3  Inquires the title of tag 3

- Remarks
  A null string ("") will be read out when there is no data in a specified tag.

- Related program code
  DAta Write title
3.4 Individual program code description

■ DA
ta Short

Setup: **DA
ta Short <param>**

- General: Register in short equalizer memory

- Parameter (when setting)
  - param: Tag number for registry in short equalizer memory
    Format: NR1
    Range: 1 to 6

- Example setup
  data short 1 Register tag 1 data in short equalizer memory

- Remarks
  After registering, the open equalizer is effective by INput Short ON.

- Related program code
  INput Short
### DA\text{ta} Template

\textbf{DA\text{ta} Template}

\begin{itemize}
  \item **Setup:** \texttt{Data Template <param1>, <param2>,...}
  \item **Query:** \texttt{?Data Template}
  \begin{itemize}
    \item (Answer) \texttt{DATA TEMPLATE <param1>, <param2>,...}
  \end{itemize}
\end{itemize}

- **General:** Sets or inquires the data transfer format

- **Parameter (when parameters are set)**
  \begin{itemize}
    \item **param1:** Data format
      \begin{itemize}
        \item Format: NR1 or string
        \begin{itemize}
          \begin{tabular}{|c|c|}
            \hline
            NR1 & String & Transfer format \\
            \hline
            0 & String & ASCII format \\
            1 & Double & IEEE Double format \\
            2 & Float & IEEE Float format \\
            3 & INVDouble & IEEE Double format in reverse byte order \\
            4 & INVFloat & IEEE Float format in reverse byte order \\
            \hline
          \end{tabular}
        \end{itemize}
      \end{itemize}
    \item **param2 or later:** Configuration and sequence of transfer data in a single block
      \begin{itemize}
        \item Format: NR1 or string
        \begin{itemize}
          \begin{tabular}{|c|c|}
            \hline
            NR1 & String & Description \\
            \hline
            1 & Sweep & Frequency \\
            2 & LOGR & Amplitude (dBV) or gain (dB) \\
            3 & R & Amplitude (Vrms) or gain \\
            4 & Theta & Phase (deg) \\
            5 & A & Real part of amplitude (Vrms) or gain \\
            6 & B & Imaginary part of amplitude (Vrms) or gain \\
            \hline
          \end{tabular}
        \end{itemize}
      \end{itemize}
  \end{itemize}

The sequence of the 2nd argument (param2) or later is optional. 6 arguments at maximum (up to param7).
3.4 Individual program code description

• Answer message (when a query is run)
  • param1: Current data format
    Format: NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NR1 String</td>
<td></td>
</tr>
<tr>
<td>0 STRING</td>
<td>Character string</td>
</tr>
<tr>
<td>1 DOUBLE</td>
<td>IEEE Double</td>
</tr>
<tr>
<td>2 FLOAT</td>
<td>IEEE Float</td>
</tr>
<tr>
<td>3 INVDOUBLE</td>
<td>IEEE Double in reverse byte order</td>
</tr>
<tr>
<td>4 INVFLOAT</td>
<td>IEEE Float in reverse byte order</td>
</tr>
</tbody>
</table>

• param2 or later: Configuration and sequence of transfer data in a single block
  Format: NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NR1 String</td>
<td></td>
</tr>
<tr>
<td>1 SWEEP</td>
<td>Frequency</td>
</tr>
<tr>
<td>2 LOGR</td>
<td>Amplitude (dBV) or gain (dB)</td>
</tr>
<tr>
<td>3 R</td>
<td>Amplitude (Vrms) or gain</td>
</tr>
<tr>
<td>4 THETA</td>
<td>Phase (deg)</td>
</tr>
<tr>
<td>5 A</td>
<td>Real part of amplitude (Vrms) or gain</td>
</tr>
<tr>
<td>6 B</td>
<td>Imaginary part of amplitude (Vrms) or gain</td>
</tr>
</tbody>
</table>

• Example setup
  data template string, sweep, logr, theta

  Sets so that the string format data will be output in sequence of frequency - string - phase

• Remarks
  The units for LOGR, R, A or B of param2 or later will depend on the current analysis mode (to be set by "DIsplay ANalysis").

<table>
<thead>
<tr>
<th>Analysis mode</th>
<th>CH1/CH2 or CH2/CH1</th>
<th>CH1 or CH2</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOGR</td>
<td>dB</td>
<td>dBV</td>
</tr>
<tr>
<td>R</td>
<td>No unit (ratio)</td>
<td>Vrms</td>
</tr>
<tr>
<td>A, B</td>
<td>No unit (ratio)</td>
<td>Vrms</td>
</tr>
</tbody>
</table>

• Related program codes
  ?DAta Read CUrrent, ?DAta Read data, DAte Write data
### DAta Write data

**Setup:**  DAta Write data <param1>, <param2>, <param3> (delimiter)(data)

- **General:** Writes data

- **Parameter (when parameters are set)**
  - **param1:** Tag number to which data is written
  - **Format:** NR1
  - **Range:** 1 to 6
  - **Default:** Current tag number
  - **param2:** Block number of data from which writing is started
  - **Format:** NR1
  - **Range:** 0 to 20000
  - **param3:** Block count of data to be written
  - **Format:** NR1
  - **Range:** 1 to 20001

- **Example setup**
  - data write data 2, 0, 101  
    
    Writes 101 data starting at the head of tag 2

- **Remarks**
  
  The transfer format etc. can be set by "DAta Template".

  The total block count in a tag to which data has been written is: param2 + param3.

  Any data following the block number specified in param2, will be not changed even after the data has been written. When the original tag is empty, any data that follows param2 will become 0.

  If an EOI is received during the transfer of some data in binary format before the specified amount of data has arrived, then the tag data will not be changed.

- **Related program codes**
  
  DAta Template, ?DAta Read Size, ?DAta Read data
**3.4 Individual program code description**

---

## DAta Write Title

Setup: **DAta Write Title <param1>, <param2>**

- **General:** Writes the title string of a tag

- **Parameter (when parameters are set)**
  - **param1:** Tag number to which the title string is written
    - Format: NR1
    - Range: 1 to 6
  - **param2:** Title string
    - Format: String, up to 63 characters

- **Example setup**
  
  data write title 2,"DATA NO.5, GAIN:10dB"

- **Remarks**
  
  When a space, comma or semicolon is contained in a title string, enclose it with double quotation codes ("), or single quotation codes (‘).

- **Related program code**
  
  ?DAta Read Title
3.4 Individual program code description

### Display ANALysis

**?DISPLAY ANALYSIS**

**Setup:**  
DIsplay ANalysis <param>

**Query:**  
?DISPLAY ANALYSIS

(Answer)  
DISPLAY ANALYSIS <param>

- **General:** Sets or inquires the analysis mode of the tag data that is being displayed (active)

- **Parameter (when parameters are set)**
  - **param:** Analysis mode
    - **Format:** NR1 or string
    - **Table:**
      | NR1 | String   | Description       |
      |-----|----------|-------------------|
      | 0   | CH1ByCH2 | Analysis mode: CH1/CH2 |
      | 1   | CH2ByCH1 | Analysis mode: CH2/CH1 |
      | 2   | CH1      | Analysis mode: CH1  |
      | 3   | CH2      | Analysis mode: CH2  |

- **Answer message (when a query is run)**
  - **param:** Current analysis mode
    - **Format:** NR1 (characters: 2) or string (to be set by SEtup Mnemonic)
    - **Table:**
      | NR1 | String   | Description       |
      |-----|----------|-------------------|
      | 0   | CH1BYCH2 | Analysis mode: CH1/CH2 |
      | 1   | CH2BYCH1 | Analysis mode: CH2/CH1 |
      | 2   | CH1      | Analysis mode: CH1  |
      | 3   | CH2      | Analysis mode: CH2  |

- **Example setup**
  
  display analysis ch1b  
  Sets the analysis mode of the currently shown tag to CH1/CH2

- **Related program code**
  
  DIsplay mode
### Display Auto

**?Display AUTO**

**Setup:** Display AUTO <param>

**Query:** ?Display AUTO

(Answer) DISPLAY AUTO <param>

- General: Sets or inquires autoscale ON/OFF

- Parameter (when parameters are set)
  - param: Setting of autoscale
    - Format: NR1 or string
      
      | NR1 | String | Description   |
      |-----|--------|--------------|
      | 0   | OFF   | Autoscale OFF|
      | 1   | ON    | Autoscale ON |

- Answer message (when a query is run)
  - param: Current setup conditions of autoscale
    - Format: NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

    | Answer format | Description   |
    |---------------|--------------|
    | 0 0 OFF       | Autoscale OFF|
    | 1 0 ON        | Autoscale ON |

- Example setup
  
  display auto on Autoscale ON

- Related program codes
  
  Display Scale Xaxis, Display Scale Y1axis, Display Scale Y2axis
3.4 Individual program code description

■ Display Grid mode

?Display Grid mode

Setup:  Display Grid mode <param>
Query:  ?Display Grid mode
        (Answer)  DISPLAY GRID MODE <param>

• General:  Sets or inquires a grid display or no grid display

• Parameter (when parameters are set)
  • param:  Grid display/no grid display
    Format:  NR1 or string
    
    | NR1 | String | Description          |
    |-----|--------|----------------------|
    | 0   | OFF    | No grid displayed    |
    | 1   | ON     | Grid displayed       |

• Answer message (when a query is run)
  • param:  Current grid display condition
    Format:  NR1 (characters: 2) or string (to be set by SEtup Mnemonic)
    Answer format
    
    | NR1 | String | Description          |
    |-----|--------|----------------------|
    | 0   | OFF    | No grid displayed    |
    | 1   | ON     | Grid displayed       |

• Example setup
  display grid on  Displays a grid

• Related program codes
  Display Grid Type, Display Grid Style
3.4 Individual program code description

- **Display Grid Type**

  **?Display Grid Type**

  Setup:  **Display Grid Type** <param>
  Query:  **?Display Grid Type**
          (Answer)  **DISPLAY GRID TYPE** <param>

  **General:** Sets or inquires the grid type

  **Parameter (when parameters are set)**
  - **param:** Grid type
    
    Format:  **NR1 or string**

    | NR1 | String | Description                  |
    |-----|--------|------------------------------|
    | 0   | Solid  | Grid in solid line          |
    | 1   | Broken | Grid in broken line         |

  **Answer message (when a query is run)**
  - **param:** Current grid type
    
    Format:  **NR1 (characters: 2) or string (to be set by SEtup Mnemonic)**

    | Answer format | Description                  |
    |---------------|------------------------------|
    | 0 SOLID       | Grid in solid line          |
    | 1 BROKEN      | Grid in broken line         |

  **Example setup**
  
  `display grid type 1`  Sets a grid in broken line

  **Related program codes**
  
  * Display Grid mode, Display Grid Style*
3.4 Individual program code description

**Display Grid Style**

**?Display Grid Style**

Setup:  **Display Grid Style <param>**

Query:  **?Display Grid Style**

(Answer)  **DISPLAY GRID STYLE <param>**

- General: Sets or inquires the grid mode

- Parameter (when parameters are set)
  - param: Grid mode
    - Format: NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>X</td>
<td>X axis only grid</td>
</tr>
<tr>
<td>1</td>
<td>XY1</td>
<td>X and Y1 axis grids</td>
</tr>
<tr>
<td>2</td>
<td>XY2</td>
<td>X and Y2 axis grids</td>
</tr>
<tr>
<td>3</td>
<td>XY1Y2</td>
<td>X, Y1 and Y2 axis grids</td>
</tr>
</tbody>
</table>

- The FRA5095 software version less than 1.99 character string designates the corresponding numerical value (NR1) for replacement.

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>F</td>
<td>Frequency axis grid only</td>
</tr>
<tr>
<td>1</td>
<td>FR</td>
<td>Frequency and gain axis grids</td>
</tr>
<tr>
<td>2</td>
<td>FTheta</td>
<td>Frequency and phase axis grids</td>
</tr>
<tr>
<td>3</td>
<td>FRTtheta</td>
<td>Frequency, gain and phase axis grids</td>
</tr>
</tbody>
</table>

- Answer message (when a query is run)
  - param: Current grid mode
    - Format: NR1 (characters: 2) or string (to be set by SETup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NR1</td>
<td>String</td>
</tr>
<tr>
<td>0</td>
<td>X</td>
</tr>
<tr>
<td>1</td>
<td>XY1</td>
</tr>
<tr>
<td>2</td>
<td>XY2</td>
</tr>
<tr>
<td>3</td>
<td>XY1Y2</td>
</tr>
</tbody>
</table>
3.4 Individual program code description

- Example setup
  
  display grid style XY1Y2  
  
  Displays a grid for X-Y1-Y2

- Remarks
  
  This setup will be valid only when the graph is a Bode diagram (in which the horizontal axis is for frequency and the vertical axis for gain/amplitude/phase).

- Related program codes
  
  DIsplay Grid mode, DIsplay Grid Type
## Display Marker Active

**Setup:**  

Display Marker Active <param>

**Query:**  

?Display Marker Active  

(Answer) DISPLAY MARKER ACTIVE <param>

- **General:** Sets or inquires the active marker

- **Parameter (when parameters are set):**
  - **param:** Setting of active marker
    - **Format:** NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Off</td>
<td>No marker displayed</td>
</tr>
<tr>
<td>1</td>
<td>Data</td>
<td>Data marker</td>
</tr>
<tr>
<td>2</td>
<td>Vertical</td>
<td>X-axis line marker</td>
</tr>
<tr>
<td>3</td>
<td>Horizontal</td>
<td>Y-axis line marker</td>
</tr>
</tbody>
</table>

- **Answer message (when a query is run):**
  - **param:** Current active marker
    - **Format:** NR1 (characters: 2) or string (to be set by SEtp Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 OFF</td>
<td>No marker displayed</td>
</tr>
<tr>
<td>1 DATA</td>
<td>Data marker</td>
</tr>
<tr>
<td>2 VERTICAL</td>
<td>X-axis line marker</td>
</tr>
<tr>
<td>3 HORIZONTAL</td>
<td>Y-axis line marker</td>
</tr>
</tbody>
</table>

- **Example setup**
  
  display ma a DATA  
  Displays a data marker, Sets to active

- **Related program code**
  
  Display Marker mode
### Display Marker mode

**Setup:** Display Marker mode <param>

**Query:** ?Display Marker mode

(Answer) DISPLAY MARKER MODE <param>

- **General:** Sets or inquires the delta marker

- **Parameter:** (when parameters are set)
  - **param:** Setting of delta marker mode
    
  **Format:** NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Normal</td>
<td>Normal marker mode</td>
</tr>
<tr>
<td>1</td>
<td>Delta</td>
<td>Delta marker mode</td>
</tr>
</tbody>
</table>

- **Answer message (when a query is run)**
  - **param:** Current delta marker mode
    
  **Format:** NR1 (characters: 2) or string (to be set by **SE**tup **M**nemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0  NORMAL</td>
<td>Normal marker mode</td>
</tr>
<tr>
<td>1  DELTA</td>
<td>Delta marker mode</td>
</tr>
</tbody>
</table>

- **Example setup**
  
  display marker mode delta Sets the delta marker mode

- **Related program code**
  
  Display Marker Active
### Display mode

**Setup:** Display mode \(<\text{param1}>, <\text{param2}>, <\text{param3}>\)

**Query:** ?Display mode

(Answer) DISPLAY MODE \(<\text{param1}>, <\text{param2}>, <\text{param3}>\)

- **General:** Sets or inquires the graph display mode
- **Parameter (when parameters are set):**
  - **\text{param1}:** X axis of graph
    - **Format:** NR1 or string
  - **\text{param2}:** Y\(_1\) axis of graph
  - **\text{param3}:** Y\(_2\) axis of graph
    - **Format:** NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LOGSweep</td>
<td>Frequency (LOG, Hz)</td>
</tr>
<tr>
<td>1</td>
<td>Sweep</td>
<td>Frequency (LIN, Hz)</td>
</tr>
<tr>
<td>2</td>
<td>LOGR</td>
<td>Gain (dB) or voltage amplitude (dBV)</td>
</tr>
<tr>
<td>3</td>
<td>R</td>
<td>Gain (no unit) or voltage amplitude (Vrms)</td>
</tr>
<tr>
<td>4</td>
<td>Theta</td>
<td>Phase (deg)</td>
</tr>
<tr>
<td>5</td>
<td>A</td>
<td>Real part of gain (no unit) or voltage amplitude (Vrms)</td>
</tr>
<tr>
<td>6</td>
<td>B</td>
<td>Imaginary part of gain (no unit) or voltage amplitude (Vrms)</td>
</tr>
<tr>
<td></td>
<td>Omit</td>
<td>None changes</td>
</tr>
</tbody>
</table>

- **\text{param2}:** Y\(_1\) axis of graph
- **\text{param3}:** Y\(_2\) axis of graph
  - **Format:** NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>RDb</td>
<td>Gain (dB) or voltage amplitude (dBV), impedance (dBΩ), admittance</td>
</tr>
<tr>
<td>3</td>
<td>R</td>
<td>Gain (no unit) or voltage amplitude (Vrms), impedance (Ω), admittance (S)</td>
</tr>
<tr>
<td>4</td>
<td>Theta</td>
<td>Phase (deg)</td>
</tr>
<tr>
<td>5</td>
<td>A</td>
<td>Real part of gain (no unit) or real part of voltage amplitude (Vrms), admittance (Ω), susceptance (S)</td>
</tr>
<tr>
<td>6</td>
<td>B</td>
<td>Imaginary part of gain (no unit) or imaginary part of voltage amplitude (Vrms), admittance (Ω), susceptance (S)</td>
</tr>
<tr>
<td>7</td>
<td>Minusb</td>
<td>-B</td>
</tr>
<tr>
<td>8</td>
<td>RLog</td>
<td>R logarithm</td>
</tr>
<tr>
<td>9</td>
<td>ALog</td>
<td>A logarithm</td>
</tr>
<tr>
<td>10</td>
<td>MALog</td>
<td>-A logarithm</td>
</tr>
<tr>
<td>11</td>
<td>BLog</td>
<td>B logarithm</td>
</tr>
<tr>
<td>12</td>
<td>MBLLog</td>
<td>-B logarithm</td>
</tr>
<tr>
<td>0</td>
<td>Off</td>
<td>No indication</td>
</tr>
<tr>
<td></td>
<td>Omit</td>
<td>No change</td>
</tr>
</tbody>
</table>
3.4 Individual program code description

- The FRA5095 software version less than 1.99 character string designates the corresponding numerical value (NR1) for replacement.

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>LOGR</td>
<td>Gain (dB) or voltage amplitude (dBV)</td>
</tr>
<tr>
<td>3</td>
<td>R</td>
<td>Gain (no unit) of voltage amplitude (Vrms)</td>
</tr>
<tr>
<td>4</td>
<td>Theta</td>
<td>Phase (deg)</td>
</tr>
<tr>
<td>5</td>
<td>A</td>
<td>Real part of gain (no unit) or real part of voltage amplitude (Vrms)</td>
</tr>
<tr>
<td>6</td>
<td>B</td>
<td>Imaginary part of gain (no unit) or imaginary part of voltage amplitude (Vrms)</td>
</tr>
<tr>
<td>7</td>
<td>Minusb</td>
<td>- imaginary part of gain or - imaginary part of voltage amplitude (Vrms)</td>
</tr>
<tr>
<td>0</td>
<td>OFF</td>
<td>No indication</td>
</tr>
</tbody>
</table>

- Answer message (when a query is run)

  - param1: X axis of current graph
  - format: NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LOGSweep</td>
</tr>
<tr>
<td>1</td>
<td>SWEEP</td>
</tr>
<tr>
<td>2</td>
<td>LOGR</td>
</tr>
<tr>
<td>3</td>
<td>R</td>
</tr>
<tr>
<td>4</td>
<td>THETA</td>
</tr>
<tr>
<td>5</td>
<td>A</td>
</tr>
<tr>
<td>6</td>
<td>B</td>
</tr>
</tbody>
</table>

  - param2: Y1 axis of current graph
  - param3: Y2 axis of current graph
  - format: NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>RDB</td>
<td>Gain (dB) or voltage amplitude (dBV), impedance (Ω), admittance (S)</td>
</tr>
<tr>
<td>3</td>
<td>R</td>
<td>Gain (no unit) or voltage amplitude (Vrms), impedance (Ω), admittance</td>
</tr>
<tr>
<td>4</td>
<td>THETA</td>
<td>Phase (deg)</td>
</tr>
<tr>
<td>5</td>
<td>A</td>
<td>Real part of gain (no unit) or real part of voltage amplitude (Vrms), impedance (Ω), conductance (S)</td>
</tr>
<tr>
<td>6</td>
<td>B</td>
<td>Imaginary part of gain (no unit) or imaginary part of voltage amplitude (Vrms), impedance (Ω), susceptance (S)</td>
</tr>
<tr>
<td>7</td>
<td>MINUSB</td>
<td>-B</td>
</tr>
<tr>
<td>8</td>
<td>RLOG</td>
<td>R logarithm</td>
</tr>
<tr>
<td>9</td>
<td>ALOG</td>
<td>A logarithm</td>
</tr>
<tr>
<td>10</td>
<td>MALOG</td>
<td>-A logarithm</td>
</tr>
<tr>
<td>11</td>
<td>BLOG</td>
<td>B logarithm</td>
</tr>
<tr>
<td>12</td>
<td>MBLOG</td>
<td>-B logarithm</td>
</tr>
<tr>
<td>0</td>
<td>OFF</td>
<td>No display</td>
</tr>
<tr>
<td></td>
<td>Omit</td>
<td>No change</td>
</tr>
</tbody>
</table>
3.4 Individual program code description

Example setup

- display mode 0, 2, 4  Sets graph to Bode diagram
  Frequency - Gain (dB) - Phase

Remarks

In the "DIsplay mode," which is intended for graph representation, the X and Y axes can be set individually by arguments, but only the following combinations are available:

<table>
<thead>
<tr>
<th>Setting Format</th>
<th>Display Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>NR1 Character String</td>
<td></td>
</tr>
<tr>
<td>0, 2, 4 LOGSweep, RDb, Theta</td>
<td>logF - dBR - θ</td>
</tr>
<tr>
<td>0, 3, 4 LOGSweep, R, Theta</td>
<td>logF - R - θ</td>
</tr>
<tr>
<td>0, 8, 4 LOGSweep, RLog, Theta</td>
<td>logF - logR - θ</td>
</tr>
<tr>
<td>1, 2, 4 Sweep, RDb, Theta</td>
<td>F - dBR - θ</td>
</tr>
<tr>
<td>1, 3, 4 Sweep, R, Theta</td>
<td>F - R - θ</td>
</tr>
<tr>
<td>1, 8, 4 Sweep, RLog, Theta</td>
<td>F - logR - θ</td>
</tr>
<tr>
<td>0, 2, 0 LOGSweep, RDb</td>
<td>logF - dBR</td>
</tr>
<tr>
<td>0, 3, 0 LOGSweep, R</td>
<td>logF - R</td>
</tr>
<tr>
<td>0, 8, 0 LOGSweep, RLog</td>
<td>logF - logR</td>
</tr>
<tr>
<td>1, 2, 0 Sweep, RDb</td>
<td>F - dBR</td>
</tr>
<tr>
<td>1, 3, 0 Sweep, R</td>
<td>F - R</td>
</tr>
<tr>
<td>1, 8, 0 Sweep, RLog</td>
<td>F - logR</td>
</tr>
<tr>
<td>0, 5, 6 LOGSweep, A, B</td>
<td>logF - A - B</td>
</tr>
<tr>
<td>0, 9, 11 LOGSweep, ALog, BLog</td>
<td>logF - logA - logB</td>
</tr>
<tr>
<td>0, 10, 11 LOGSweep, MALog, BLog</td>
<td>logF - log(-A) - log(-B)</td>
</tr>
<tr>
<td>0, 9, 12 LOGSweep, ALog, MBLog</td>
<td>logF - logA - log(-B)</td>
</tr>
<tr>
<td>0, 10, 12 LOGSweep, MALog, MBLog</td>
<td>logF - log(-A) - log(-B)</td>
</tr>
<tr>
<td>1, 5, 6 Sweep, A, B</td>
<td>F - A - B</td>
</tr>
<tr>
<td>1, 9, 11 Sweep, ALog, BLog</td>
<td>F - logA - logB</td>
</tr>
<tr>
<td>1, 10, 11 Sweep, MALog, BLog</td>
<td>F - log(-A) - logB</td>
</tr>
<tr>
<td>1, 9, 12 Sweep, ALog, MBLog</td>
<td>F - logA - log(-B)</td>
</tr>
<tr>
<td>1, 10, 12 Sweep, MALog, MBLog</td>
<td>F - log(-A) - log(-B)</td>
</tr>
<tr>
<td>0, 4, 0 LOGSweep, Theta</td>
<td>logF - θ</td>
</tr>
<tr>
<td>1, 4, 0 Sweep, Theta</td>
<td>F - θ</td>
</tr>
<tr>
<td>4, 2, 0 Theta, RDb</td>
<td>θ - dBR</td>
</tr>
<tr>
<td>4, 3, 0 Theta, R</td>
<td>θ - R</td>
</tr>
<tr>
<td>4, 8, 0 Theta, RLog</td>
<td>θ - logR</td>
</tr>
<tr>
<td>5, 6, 0 A, B</td>
<td>a - b</td>
</tr>
<tr>
<td>5, 7, 0 A, Minusb</td>
<td>a - -b</td>
</tr>
</tbody>
</table>
The units for dBR, R, A, B and Minusb will be as follows according to the current setup of the analysis mode ("DIsplay ANalysis").

- **When DIsplay Units are set to gain**

<table>
<thead>
<tr>
<th>Analysis mode &quot;DIsplay Analysis&quot;</th>
<th>CH1/CH2 or CH2/CH1</th>
<th>CH1 or CH2</th>
</tr>
</thead>
<tbody>
<tr>
<td>dBR Gains</td>
<td>Voltage amplitude (dBV)</td>
<td></td>
</tr>
<tr>
<td>R Gain (no unit)</td>
<td>Voltage amplitude (Vrms)</td>
<td></td>
</tr>
<tr>
<td>A Real part of gain (no unit)</td>
<td>Real part of voltage amplitude (Vrms)</td>
<td></td>
</tr>
<tr>
<td>B Imaginary part of gain (no unit)</td>
<td>Imaginary part of voltage amplitude (Vrms)</td>
<td></td>
</tr>
<tr>
<td>Minusb -Imaginary part of gain (no unit)</td>
<td>-Imaginary part of voltage amplitude (Vrms)</td>
<td></td>
</tr>
</tbody>
</table>

- **When DIsplay Units are set to impedance**

<table>
<thead>
<tr>
<th>Analysis mode &quot;DIsplay ANalysis&quot;</th>
<th>CH1/CH2</th>
<th>CH2/CH1</th>
<th>CH1</th>
<th>CH2</th>
</tr>
</thead>
<tbody>
<tr>
<td>dBR Impedance (dBΩ)</td>
<td>Admittance (dBS)</td>
<td>Voltage amplitude (dBV)</td>
<td>Current amplitude (dBA)</td>
<td></td>
</tr>
<tr>
<td>R Impedance (Ω)</td>
<td>Admittance (S)</td>
<td>Voltage amplitude (Vrms)</td>
<td>Current amplitude (Arms)</td>
<td></td>
</tr>
<tr>
<td>A Resistance (Ω)</td>
<td>Conductance (S)</td>
<td>Real part of voltage amplitude (Vrms)</td>
<td>Real part of current amplitude (Arms)</td>
<td></td>
</tr>
<tr>
<td>B Reactance (Ω)</td>
<td>Susceptance (S)</td>
<td>Imaginary part of voltage amplitude (Vrms)</td>
<td>Imaginary part of current amplitude (Arms)</td>
<td></td>
</tr>
</tbody>
</table>

- Related program code

DIsplay ANalysis
3.4 Individual program code description

**Display Phase**

**?Display Phase**

Setup:  **Display Phase** <param>

Query:  **?Display Phase**  
(Answer)  **DISPLAY PHASE** <param>

- **General**: Sets or inquires the phase display range

- **Parameter (when parameters are set)**
  - **param**: Central value in the phase display range
  - **Format**: NR3
  - **Range**: -180, 0, 180 (deg)

<table>
<thead>
<tr>
<th>param set value</th>
<th>Phase display range</th>
</tr>
</thead>
<tbody>
<tr>
<td>-180</td>
<td>-360 to 0 deg</td>
</tr>
<tr>
<td>0</td>
<td>-180 to 180 deg</td>
</tr>
<tr>
<td>180</td>
<td>0 to 360 deg</td>
</tr>
</tbody>
</table>

- **Answer message (when a query is run)**
  - **param**: Central value in the current phase display range
  - **Format**: NR3
  - **Range**: -180E+00 to +180E+00
  
  (Available values: 180, -0, 180)

  **Characters**: 11

- **Example setup**
  
  display phase 0  

  Sets the phase display range to -180 to +180 deg

- **Remarks**

  Even when the graph is set to autoscale, the phase display range defined here will not be exceeded.
3.4 Individual program code description

**Display Scale Xaxis**

**?Display Scale Xaxis**

Setup:  Display Scale Xaxis <param1>, <param2>
Query:  ?Display Scale Xaxis

(Answer)  DISPLAY SCALE XAXIS <param1>, <param2>

- **General:** Sets or inquires the display range for X axis of graph

- **Parameter (when parameters are set):**
  - param1:  Maximum display range for X axis of graph
  - param2:  Minimum display range for X axis of graph

- **Format:**  NR3

- **Answer message (when a query is run):**
  - param1:  Maximum display range for X axis of current graph
  - param2:  Minimum display range for X axis of current graph

- **Format:**  NR3

- **Example setup**
  
  display scale x 1, 1e6  
  Sets the X-axis display range to 1 to 1e6
  1 Hz to 1 MHz if the display mode of X axis is set to Frequency

- **Remarks**

The setup or query of the maximum or minimum display range for the X axis using this program code, will depend on the display mode of the current graph (to be set by "Display mode").

<table>
<thead>
<tr>
<th>Display mode for X axis</th>
<th>Unit</th>
<th>When set</th>
<th>When queried</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Setup range</td>
<td>Characters</td>
<td></td>
</tr>
<tr>
<td>Frequency F, logF Hz</td>
<td>Hz</td>
<td>0.1E-3 to 15E+6</td>
<td>17</td>
</tr>
<tr>
<td>Phase ( \theta )</td>
<td>deg</td>
<td>-1.0E9 to 1.0E9</td>
<td>11</td>
</tr>
<tr>
<td>Real part (gain or amplitude) a</td>
<td>No unit or Vrms</td>
<td>-1.0E9 to 1.0E9</td>
<td>11</td>
</tr>
<tr>
<td>(Resistance, conductance) A</td>
<td>( \Omega ) or S</td>
<td>-1.0E9 to 1.0E9</td>
<td>11</td>
</tr>
</tbody>
</table>

The numeric value obtained in response to a query will be the one set in the menu [Graph][SCALE], rather than the display range of the X axis that is currently shown in the graph.
3.4 Individual program code description

Display Scale Y1axis

?Display Scale Y1axis

Setup: Display Scale Y1axis <param1>, <param2>
Query: ?Display Scale Y1axis
(Answer) DISPLAY SCALE Y1AXIS <param1>, <param2>

- General: Sets or inquires the display range for Y1 axis of graph

- Parameter (when parameters are set)
  - param1: Maximum display range for Y1 axis of graph
  - param2: Minimum display range for Y1 axis of graph
  Format: NR3

- Answer message (when a query is run)
  - param1: Maximum display range for Y1 axis of current graph
  - param2: Minimum display range for Y1 axis of current graph
  Format: NR3

- Example setup
  display scale y1 0, 20
  Sets the Y1-axis display range to 0 to 20. 0 dB to 20 dB if the display mode of Y1 axis is set to Gain (dB)

- Remarks
  The setup or query of the maximum or minimum display range or Y1 axis using this program code, will depend on the display mode of the current graph (to be set by "Display mode").

<table>
<thead>
<tr>
<th>Display mode for Y1 axis</th>
<th>Unit</th>
<th>When set</th>
<th>When queried</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Setup range</td>
<td>Characters</td>
</tr>
<tr>
<td>Gain or amplitude</td>
<td>dB or dBV</td>
<td>-1000 to 1000</td>
<td>12</td>
</tr>
<tr>
<td>R, dBR</td>
<td>No unit or Vrms</td>
<td>-1.0E9 to 1.0E9</td>
<td>10</td>
</tr>
<tr>
<td>Phase θ</td>
<td>deg</td>
<td>-1.0E9 to 1.0E9</td>
<td>11</td>
</tr>
<tr>
<td>Imaginary part (gain</td>
<td>No unit or Vrms</td>
<td>-1.0E9 to 1.0E9</td>
<td>11</td>
</tr>
<tr>
<td>or amplitude) b, -b</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Impedance, admittance)</td>
<td>dBΩ or dBΣ</td>
<td>-1000 to 1000</td>
<td>12</td>
</tr>
<tr>
<td>R, logR, dBR</td>
<td>Ω or S</td>
<td>-1.0E9 to 1.0E9</td>
<td>10</td>
</tr>
<tr>
<td>(Resistance, conductance)</td>
<td>Ω or S</td>
<td>-1.0E9 to 1.0E9</td>
<td>11</td>
</tr>
<tr>
<td>A, logA, log (-A)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The numeric value obtained in response to a query will be the one in the menu GraphSCALE, rather than the display range of the Y1 axis that is currently shown in the graph.
### Display Scale Y2axis

**?Display Scale Y2axis**

Setup:  

```text
Display Scale Y2axis <param1>, <param2>
```

Query:  

```text
?Display Scale Y2axis
```

(Answer)  

```text
DISPLAY SCALE Y2AXIS <param1>, <param2>
```

- **General:** Sets or inquires the display range for Y-2 axis of graph

- **Parameter (when parameters are set):**
  - `param1:` Maximum display range for Y-2 axis of graph
  - `param2:` Minimum display range for Y-2 axis of graph

  **Format:** NR3

- **Answer message (when a query is run):**
  - `param1:` Maximum display range for Y-2 axis of current graph
  - `param2:` Minimum display range for Y-2 axis of current graph

  **Format:** NR3

- **Example setup:**

  ```text
display scale y2 -180, 180
```

  Sets the Y-2-axis display range to -180 to +180 ±180 deg if the display mode of Y-2 axis is set to Phase

- **Remarks:**

  The setup or query of the maximum or minimum display range of the Y-2 axis using this program code, will depend on the display mode of the current graph (to be set by "Display mode"). Only the phase can be shown on the Y-2 axis.

<table>
<thead>
<tr>
<th>Display mode for Y-2 axis</th>
<th>Unit</th>
<th>When set (Setup range)</th>
<th>When queried (Characters)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase θ</td>
<td>deg</td>
<td>-1.0E9 to 1.0E9</td>
<td>11</td>
</tr>
<tr>
<td>(None displayed)</td>
<td>–</td>
<td>–</td>
<td>11</td>
</tr>
<tr>
<td>(Reactance, susceptance)</td>
<td>Ω or S</td>
<td>-1.0E9 to 1.0E9</td>
<td>11</td>
</tr>
</tbody>
</table>

  The numeric value obtained in response to a query will be the one set in the menu GraphSCALE, rather than the display range of the Y-2 axis that is currently being shown in the graph. However, if a query is run while the Y-2 axis is not shown, then an answer will be returned with both `param1` and `param2` as 0.0E+0.
3.4 Individual program code description

- **Display Units**

  **?Display Units**

  **Setup:**  Display Units <param>
  **Query:**  ?Display Units
  
  (Answer)  DISPLAY UNITS <param>

  - General: Display units setup

  - Parameter (when parameters are set)
    - param: Display units
      - Format: NR1 or character string

        | NR1 | String  | Description   |
        |-----|---------|---------------|
        | 0   | Gain    | Gain display  |
        | 1   | Impedance | Ohms (Ω or S) display |

  - Answer message (when a query is run)
    - param: Present graph display window mode
      - Format: NR1 (character string: 2) or SEtup Mnemonic

        | Answer format | String    | Description   |
        |---------------|-----------|---------------|
        | NR1           | GAIN      | Gain display  |
        | 0             | IMPEDANCE | Ohms (Ω or S) display |

  - Example setup
    - display units 1

  - Remarks
    - Display units differ between graph and analysis modes.

  - Related program code
    - INput Open
3.4 Individual program code description

- **Display Window**

  **?Display Window**

  Setup:  `Display Window <param>`
  Query:  `?Display Window`  
  (Answer)  `DISPLAY WINDOW <param>`

  - General: Sets or queries the graph display window mode

  - Parameter (when parameters are set)
    - `param`: Setting of graph display window mode
      Format:  NR1 or string
      | NR1 | String | Description |
      |-----|--------|------------|
      | 0   | Single | Single view |
      | 1   | Split  | Split view |

  - Answer message (when a query is run)
    - `param`: Current graph display window mode
      Format:  NR1 (characters: 2) or string (to be set by `SEtup Mnemonic`)
      Answer format
      | NR1 | String | Description |
      |-----|--------|------------|
      | 0   | SINGLE | Single view |
      | 1   | SPLIT  | Split view |

  - Example setup
    `display window split`  Sets a graph to the SPLIT view mode (split it vertically into two parts)
3.4 Individual program code description

■ File DElete Disk

Setup:  File DElete Disk <param>

- General: Deletes files

- Parameter (when parameters are set)
  - param: Name of a file to be deleted
    Format: String
    Range: Name strings of files that exist in a USB flash drive

- Example setup
  
  file delete disk data1.dat  Deletes the data file "data1.dat"

- Remarks
  
  An error will result if you specify the name of a file that does not exist.

- Related program code
  
  ?File Dir Disk
3.4 Individual program code description

## File DElete Mass

**Setup:**  \textbf{File DElete Mass <param>}

- **General:** Deletes mass memory

- **Parameter (when parameters are set)**
  - **param:** Number of mass memory to be deleted
    - **Format:** NR1
    - **Range:** 1 to (a number that exists in mass memory)

- **Example setup**
  
  \texttt{file delete mass 21}  \hspace{1cm} Deletes mass memory #21

- **Remarks**
  
  An error will result if you specify the number of mass memory that does not exist.

- **Related program code**
  
  ?File Dir Mass
3.4 Individual program code description

- File DElete Permanent

Setup: File DElete Permanent <param>

- General: Deletes permanent memory

- Parameter (when parameters are set)
  - param: Number of permanent memory to be deleted
    - Format: NR1
    - Range: 1 to (a number that exists in permanent memory)

- Example setup
  
  ```
  file delete mass 2
  Deletes permanent memory #2
  ```

- Remarks
  An error will result if you specify the number of permanent memory that does not exist.

- Related program code
  
  `?File DIr Permanent`
3.4 Individual program code description

- **?File Dir Disk**

Query: **?File Dir Disk**

(Answer) <param1>, <param2>, <param3> block delimiter

- **General**: Outputs the list of files that are stored on a USB flash drive

- **Answer message (when a query is run)**
  - **param1**: File name
    Format: Character string: filename (8-characters) + " " + filename extension (3characters)
    The characters in a file name and an extension that contains less than 8 and 3 characters respectively, will be placed to left and spaces will fill in for the missing characters.
  - **param2**: Date on which a file was created
    Format: String YYYY-MM-DD (year-month-day)
  - **param3**: Time at which a file was created
    Format: String HH:MM:SS (hour:minute:second) SS is constantly "00"

Individual file names are divided by block delimiters, and the final one is followed by a delimiter.

- **Example setup**
  
  `?file dir disk`

- **Related program codes**
  
  `?File Dir Mass, ?File Dir Permanent`
### ?File Dlr Mass

**Query:** ?File Dlr Mass  
(Answer) <param1>, <param2>, <param3> (block delimiter)

- **General:** Outputs the list of file numbers that are registered with mass memory

- **Answer message (when a query is run)**
  - **param1:** Registration number of mass memory  
    Format: NR1 (characters:5)  
    Range: Registration numbers that exist in mass memory
  - **param2:** Date on which a file was created  
    Format: String YYYY-MM-DD (year-month-day)
  - **param3:** Time at which a file was created  
    Format: String HH:MM:SS (hour:minute:second) SS is constantly "00"

Individual file names are divided by block delimiters, and the final one is followed by a delimiter.

- **Example setup**
  
  ?file dir mass

- **Related program codes**
  
  ?File Dlr Disk, ?File Dlr Permanent
### 3.4 Individual program code description

- **?File Dlr Permanent**

  **Query:**  
  
  ?File Dlr Permanent  
  
  (Answer)  
  
  <param1>, <param2>, <param3> (block delimiter)  
  
  :  

  - **General:** Outputs the list of file numbers that are registered with permanent memory

  - **Answer message (when a query is run)**
    
    - **param1:** Registration number of permanent memory  
      
      Format: NR1 (characters:5)  
      
      Range: Registration numbers that exist in permanent memory  
    
    - **param2:** Date on which a file was created  
      
      Format: String YYYY-MM-DD (year-month-day)  
    
    - **param3:** Time at which a file was created  
      
      Format: String HH:MM:SS (hour:minute:second) SS is constantly "00"  

  Individual file names are divided by block delimiters, and the final one is followed by a delimiter.

  - **Example setup**
    
    ?file dir permanent

  - **Related program codes**
    
    ?File Dlr Disk, ?File Dlr Mass
### File Load Disk data

**Setup:** File Load Disk data `<param1>`, `<param2>`

- **General:** Loads data files from a USB flash drive

- **Parameter (when parameters are set)**
  - **param1:** Name of a data file to be loaded (it must include the extension ".dat")
    - **Format:** String
    - **Range:** Filename strings that exist in a USB flash drive
  - **param2:** Tag number to which data is written
    - **Format:** NR1
    - **Range:** 1 to 6
    - **omit:** Current tag number

- **Example setup**
  
  ```
  file load disk data data1.dat
  ```

  Loads the data file "data1.dat"

- **Remarks**
  
  An error will result if an attempt is made to load a file that does not exist. Be sure to add ".dat" to a file name as an extension.

- **Related program code**
  
  File Save Disk data
3.4 Individual program code description

### File Load Disk Condition

**Setup:**  
**File Load Disk Condition** <param>

- **General:** Loads setup condition files from a USB flash drive

- **Parameter (when parameters are set)**
  - **param:** Name of a setup condition file to be loaded (it must include the extension ".con")
    - **Format:** String
    - **Range:** Filename strings that exist in a USB flash drive

- **Example setup**
  
  ```plaintext
  file load disk condition setup1.con  Loads the setup condition file "setup1.con"
  ```

- **Remarks**
  An error will result if an attempt is made to load a file that does not exist.
  Be sure to add ".con" to a file name as an extension.
  Once a setup condition file has been loaded, the FRA5097 will follow the information in that file for its settings.

- **Related program code**
  
  **File Save Disk Condition**
3.4 Individual program code description

■ File Load Mass

Setup:  File Load Mass <param1>, <param2>

- General: Loads mass data and assigns it to a tag

- Parameter (when parameters are set)
  - param1:  File numbers that are registered with mass memory
    Format: NR1
    Range: File numbers that exist in mass memory
  - param2:  Tag number to which data is written
    Format: NR1
    Range: 1 to 6
    Omit: Current tag number

- Example setup
  file load mass 3, 1  Assigns mass memory #3 to tag 1

- Remarks
  An error will result if an attempt is made to load the number of mass memory that does not exist.
  The list of file numbers registered with mass memory can be read by "?File Dir Mass".

- Related program codes
  ?File Dir Mass, File Save Mass
### File Load Permanent

**Setup:**  **File Load Permanent** <param1>, <param2>

- **General:** Loads permanent data and assigns it to a tag

- **Parameter (when parameters are set)**
  - **param1:** File numbers that are registered with permanent memory
    - **Format:** NR1
    - **Range:** File numbers that exist in permanent memory
  - **param2:** Tag number to which data is written
    - **Format:** NR1
    - **Range:** 1 to 6
    - **Omit:** Current tag number

- **Example setup**
  
  file load permanent 15, 2  Assigns permanent memory #15 to tag 2

- **Remarks**
  An error will result if an attempt is made to load the number of permanent memory that does not exist.
  The list of file numbers registered with permanent memory can be read by "?File Dlr Permanent".

- **Related program codes**
  
  ?File Dlr Permanent, File Save Permanent
3.4 Individual program code description

File Rename Disk

Setup: File Rename Disk <param1>, <param2>

- General: Renames files in a USB flash drive

- Parameter (when parameters are set)
  - param1: Current file name
    Format: String
    Range: Names of files that exist on a USB flash drive (including extensions)
  - param2: New file name
    Format: String
    Range: Names of files that do not exist on a USB flash drive (including extensions)

- Example setup
  file rename disk nowdata.dat old1.dat
  Renames the file "nowdata.dat" on a USB flash drive into "old1.dat"

- Remarks
  An error will result if you specify the name of a file not existing in param1 nor the name of a file existing in a USB flash drive in param2.
  The list of files disk can be read by "?File Dlr Disk".

- Related program codes
  ?File Dlr Disk, File DElete Disk
### File Save Disk data

**Setup:**  
**File Save Disk data** <param1>, <param2>

- **General:** Saves tag data in a USB flash drive

- **Parameter (when parameters are set)**
  - **param1:** Tag number of data to be saved  
    - Format: NR1  
    - Range: 1 to 6  
    - omit: Current tag number
  - **param2:** Name of a file to be saved, including the extension ".dat"  
    - Format: String

- **Example setup**
  
  ```
  file save disk data 1, data1.dat
  ```

  Saves tag 1 data to a USB flash drive as "data1.dat"

- **Remarks**
  
  If another file with an identical name is already in the USB flash drive, then it will be overwritten, so that the previous file data will be cleared.

- **Related program codes**
  
  File Save Disk Condition, File Load Disk data, ?File Dir Disk
3.4 Individual program code description

- **File Save Disk Condition**

**Setup:** File Save Disk Condition <param>

- **General:** Saves setup conditions to a USB flash drive

- **Parameter (when parameters are set)**
  - **param:** Name of a file to be saved, including the extension ".con"
    - **Format:** String

- **Example setup**
  
  file save disk condition setup1.con Saves the current setup conditions to a USB flash drive as "setup1.con"

- **Remarks**
  If another file with an identical name is already present in the USB flash drive, it will be overwritten, so that the previous file data will be cleared.

- **Related program codes**
  File Save Disk data, File Load Disk data, ?File Dir Disk
### File Save Mass

**Setup:** File Save Mass <param1>, <param2>

- **General:** Stores tag data to mass memory

- **Parameter (when parameters are set)**
  - **param1:** Tag number to which tag data is stored
    - Format: NR1
    - Range: 1 to 6
    - omit: Current tag number
  - **param2:** Mass memory number to which data is stored
    - Format: NR1
    - Range: 1 to 999
    - omit: Generates a memory number automatically

- **Example setup**
  
  ```
  file save mass 3, 5
  ```
  Stores tag 3 data to mass memory #5

- **Remarks**
  Unique mass memory and permanent memory numbers are assigned to ensure that they are not identical.

  If a mass memory number in which data is to be stored already exists, the data will be overwritten and cleared. Similarly, if some data is written in a mass memory number identical to a permanent memory number that already exists, the permanent memory data will be deleted.

  If param2 is omitted, the subsequent number of the last memory number in which data was stored, will be generated automatically.

  If some data already exists in the mass or permanent memory of that number, the data will be overwritten.

- **Related program codes**
  
  File Load Mass, ?File Dir Mass
3.4 Individual program code description

File Save Permanent

Setup: File Save Permanent <param1>, <param2>

• General: Stores tag data in permanent memory

• Parameter (when parameters are set)
  • param1: Tag number to which tag data is stored
    Format: NR1
    Range: 1 to 6
    omit: Current tag number
  • param2: Permanent memory number to which tag data is stored
    Format: NR1
    Range: 1 to 999
    omit: Generates a memory number automatically

• Example setup
  file save permanent 2, 1 Stores tag 2 data to permanent memory #1

• Remarks
  Unique mass memory and permanent memory numbers will be assigned to ensure that they are not identical.
  If a permanent memory number to which data is to be stored already exists, the data will be overwritten and cleared. Similarly, if some data is written in a permanent memory number identical to a mass memory number that already exists, the mass memory data item will be deleted.
  If param2 is omitted, the subsequent number of the last memory number to which data was stored, will be generated automatically. If some data already exists in the mass or permanent memory of that number, the data will be overwritten.

• Related program codes
  File LoadPermanent, ?File Dir Permanent
3.4 Individual program code description

- **INput Action**

  ?INput Action

  Setup: INput Action <param1>, <param2>
  Query: ?INput Action
           (Answer) INPut ACtion <param1>, <param2>

  - General: Sets or inquires the action taken when overload input is detected
  - Parameter (when parameters are set)
    - **param1:** Buzzer action when overload input is detected
      Format: NR1 or string
      | NR1 | String | Description          |
      |-----|--------|----------------------|
      | 0   | OFF    | No buzzer sounds     |
      | 1   | ON     | Buzzer sounds        |
      Omit: None changes
    - **param2:** Oscillator and sweep when overload input is detected
      Format: NR1 or string
      | NR1 | String   | Description            |
      |-----|----------|------------------------|
      | 0   | OFF      | Oscillation & sweep continue |
      | 1   | SWEEP    | Only sweep stops       |
      | 2   | OSC      | Oscillation & sweep stop |
      Omit: None changes
  - Answer message (when a query is run)
    - **param1:** Current buzzer action when overload input is detected
      Format: NR1 (characters: 2) or string (to be set by SEtup Mnemonic)
      | Answer format | Description          |
      | NR1 | String          |
      | 0   | OFF             | No buzzer sounds     |
      | 1   | ON              | Buzzer sounds        |
    - **param2:** Current oscillator and sweep when overload input is detected
      Format: NR1 (characters: 2) or string (to be set by SEtup Mnemonic)
      | Answer format | Description            |
      | NR1 | String           |
      | 0   | OFF              | Oscillation & sweep continue |
      | 1   | SWEEP            | Only sweep stops       |
      | 2   | OSC              | Oscillation & sweep stop |
  - Example setup
    INput action ON,OSC
    Stops both oscillation and sweep when overload input is detected
## INput CAlculate

### ?INput CAlculate

**Setup:** `INput CAlculate <param1>, <param2>, <param3>`

**Query:** `?INput CAlculate`

(Answer) `INPUT CALCULATE <param1>, <param2>, <param3>`

- **General:** Sets or inquires the input weights

- **Parameter (when parameters are set)**
  - **param1:** Number to be applied to CH1 input
    - Format: NR3
    - Range: 0.0 to 1.0E+6
    - Omit: None changes
  - **param2:** Number to be applied to CH2 input
    - Format: NR3
    - Range: 0.0 to 1.0E+6
    - Omit: None changes
  - **param3:** Phase invert
    - Format: NR1 or character string
      - NR1 String Description
        | 0 | OFF | Phase not inverted |
        | 1 | ON  | Phase inverted |
    - Omit: None changes

- **Answer message (when a query is run)**
  - **param1:** Number to be applied to the current CH1 input
    - Format: NR3
    - Range: 0.00E+00 to 1.00E+06
    - Characters: 9
  - **param2:** Number to be applied to the current CH2 input
    - Format: NR3
    - Range: 0.00E+00 to 1.00E+06
    - Characters: 9
  - **param3:** Phase invert
    - Format: NR1 (character string: 2) or SEtup Mnemonic
      - Answer format
        | NR1 | String | Description |
        |-----|--------|-------------|
        | 0   | OFF   | Phase not inverted |
        | 1   | ON    | Phase inverted |

- **Example setup**
  - `input calculate 0.5, 1.0`  
    CH1 factor: 0.5, CH2 factor: 1.0
3.4 Individual program code description

**INput Equalize**

**?INput Equalize**

Setup:  **INput Equalize <param>**
Query:  **?INput Equalize**  
       (Answer)  **INPUT EQUALIZE <param>**

- **General:** Sets or inquires the input equalization process

- **Parameter (when parameters are set)**
  - **param:** ON/OFF status of equalization
  - **Format:** NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td><strong>OFF</strong></td>
<td>Not equalized</td>
</tr>
<tr>
<td>1</td>
<td><strong>ON</strong></td>
<td>Equalized</td>
</tr>
</tbody>
</table>

- **Answer message (when a query is run)**
  - **param:** ON/OFF status of current equalization
  - **Format:** NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NR1</td>
<td>String</td>
</tr>
<tr>
<td>0</td>
<td><strong>OFF</strong></td>
</tr>
<tr>
<td>1</td>
<td><strong>ON</strong></td>
</tr>
</tbody>
</table>

- **Example setup**
  - input equalize on  Performs equalization

- **Related program code**
  - DAData Equalize
3.4 Individual program code description

- **INput Open**

  ?INput Open

  Setup:  **INput Open** <param>
  Query:  ?INput Open
          (Answer)  **INPUT OPEN** <param>

  - General: Open equalize function setting/query

  - Parameter (when setting)
    - param: Open equalize on/off
    - Format: NR1 or character string
      
      | NR1 | String | Description               |
      |-----|--------|---------------------------|
      | 0   | **OFF** | Without open correction  |
      | 1   | **ON**  | Open equalized            |

  - Response message (when query sent)
    - param: Present open equalize on/off state
    - Format: NR1 (character string: 2) or character string SEtup Mnemonic
      
      | Answer format | Description               |
      |---------------|---------------------------|
      | 0 OFF         | Without open correction  |
      | 1 ON          | Open equalized            |

  - Usage example
    input open on  With open correction

  - Related program code
    DAAta Open
### 3.4 Individual program code description

#### INput OVer

**?INput OVer**

Setup:  **INput OVer** <param1>, <param2>

Query:  **?INput OVer**

(Answer)  **INPUT OVER** <param1>, <param2>

- **General:** Sets or inquires the detection level for overload input

- **Parameter (when parameters are set)**
  - **param1:** Detection level for CH1 overload
    - Format: NR3
    - Range: 0 to 250 (Vrms)
    - Omit: None changes
  - **param2:** Detection level for CH2 overload
    - Format: NR3
    - Range: 0 to 250 (Vrms)
    - Omit: None changes

- **Answer message (when a query is run)**
  - **param1:** Current detection level for CH1 overload
    - Format: NR3
    - Range: 0.00E+00 to 250E+00
    - Characters: 9
  - **param2:** Current detection level for CH2 overload
    - Format: NR3
    - Range: 0.00E+00 to 250E+00
    - Characters: 9

- **Example setup**

  input over 123, 10e-3  
  CH1: 123 Vrms, CH2: 10 mVrms

  ?input over

  (Answer)  123E+00, 10E-03
3.4 Individual program code description

■ INput Short

?INput Short

Setup:    INput Short <param>
Query:    ?INput Short
           (Answer)   INPUT SHORT <param>

● General: Short correction function set/query

● Parameter (when setting)
  • param:     Short function on/off
    Format:    NR1 or character string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>OFF</td>
<td>Without short correction</td>
</tr>
<tr>
<td>1</td>
<td>ON</td>
<td>With short correction</td>
</tr>
</tbody>
</table>

● Answer message (when a query is run)
  • param:     Present short on/off state
    Format:    NR1 (character string: 2) or character string SEtup Mnemonic setting

<table>
<thead>
<tr>
<th>Answer format</th>
</tr>
</thead>
<tbody>
<tr>
<td>NR1</td>
</tr>
<tr>
<td>------</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

● Example setup
  input short on       Short is corrected

● Related program code
  DAta Short
3.4 Individual program code description

■ Measure Auto MAx Cycle

?Measure Auto MAx Cycle

Setup: Measure Auto MAx Cycle <param>
Query: ?Measure Auto MAx Cycle
(Answer) MEASURE AUTO MAX CYCLE <param>

● General: Sets or inquires the maximum cycles of automatic integration

● Parameter (when parameters are set)
  • param: Maximum cycles of integration
    Format: NR1
    Range: 2 to 9999 (cycles)

● Answer message (when a query is run)
  • param: Current maximum cycles of integration
    Format: NR1
    Range: 2 to 9999
    Characters: 5

● Example setup
  measure auto max cycle 100  Max cycles of automatic integration: 100

● Remarks
  This setup will not be valid unless "Measure Auto MAx Type" is set to 0 or Cycle.

● Related program codes
  Measure Auto MAx Sec, Measure Auto MAx Type
3.4 Individual program code description

Measure Auto MAx Sec

?Measure Auto MAx Sec

Setup: Measure Auto MAx Sec <param>
Query: ?Measure Auto MAx Sec
   (Answer) MEASURE AUTO MAX SEC <param>

- General: Sets or inquires the maximum time of automatic integration

- Parameter (when parameters are set)
  - param: Maximum integration time
    Format: NR3
    Range: 0 to 9999 (sec)

- Answer message (when a query is run)
  - param: Current maximum integration time
    Format: NR3
    Range: 0.00000E+00 to 9.99900E+03
    Characters: 12

- Example setup
  measure auto max sec 1  Max time of automatic integration: 1 second

- Remarks
  Two cycles of integration will definitely be run regardless of the setup. Therefore, even when the integration time is set to 0 second, the measuring time will be two seconds or more at a measuring frequency of 1 Hz.
  This setup will not be valid unless "Measure Auto MAx Type" is set to 1 or Sec.

- Related program codes
  Measure Auto MAx Cycle, Measure Auto MAx Type
### 3.4 Individual program code description

#### Measure Auto MAX Type

**?Measure Auto MAX Type**

**Setup:**  
Measure Auto MAX Type <param>

**Query:**  
?Measure Auto MAX Type

(Answer)  
MEASURE AUTO MAX TYPE <param>

- **General:** Sets or inquires the setting type of maximum automatic integration

- **Parameter (when parameters are set)**
  - **param:** Setting type of maximum integration
  
  **Format:** NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Cycle</td>
<td>Set by a number of times (cycles)</td>
</tr>
<tr>
<td>1</td>
<td>Sec</td>
<td>Set by time</td>
</tr>
</tbody>
</table>

- **Answer message (when a query is run)**
  - **param:** Current setting type of maximum integration
  
  **Format:** NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 CYCLE</td>
<td>Set by a number of times (cycles)</td>
</tr>
<tr>
<td>1 SEC</td>
<td>Set by time</td>
</tr>
</tbody>
</table>

- **Example setup**

  measure auto max type cycle  
  Sets automatic integration by a number of times (cycles)

- **Remarks**
  
  Although the maximum level of automatic integration can be set individually using "Measure Auto MAX Cycle" and "Measure Auto MAX Sec", it is the automatic integration set by this program code that will actually be used.

- **Related program codes**

  Measure Auto MAX Cycle, Measure Auto MAX Sec
3.4 Individual program code description

- **Measure Auto mode**

  ?Measure Auto mode

  Setup:  Measure Auto mode <param>
  Query:  ?Measure Auto mode  
  (Answer) MEASURE AUTO MODE <param>

  - General: Sets or inquires the action of automatic integration

  Parameter (when parameters are set)
  
  - param: Action of automatic integration
  
  Format: NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Off</td>
<td>Manual integration mode</td>
</tr>
<tr>
<td>1</td>
<td>Short</td>
<td>SHORT mode</td>
</tr>
<tr>
<td>2</td>
<td>Long</td>
<td>LONG mode</td>
</tr>
</tbody>
</table>

  - Answer message (when a query is run)
  
  - param: Action of current automatic integration
  
  Format: NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 OFF</td>
<td>Manual integration mode</td>
</tr>
<tr>
<td>1 SHORT</td>
<td>SHORT mode</td>
</tr>
<tr>
<td>2 LONG</td>
<td>LONG mode</td>
</tr>
</tbody>
</table>

  - Example setup
  
  measure auto mode SHORT  
  Set the automatic integration of SHORT mode

  - Related program code
  
  Measure Coherence
3.4 Individual program code description

- **Measure Coherence**

  ?Measure Coherence

  **Setup:** Measure Coherence <param>
  **Query:** ?Measure Coherence
  (Answer) MEASURE COHERENCE <param>

  - General: Sets or inquires the coherence calculation mode

  - Parameter (when parameters are set)
    - param: Coherence calculation mode
      - Format: NR1 or string
        - | NR1 | String | Description |
        - |-----|--------|-------------|
        - | 0   | Off    | OFF         |
        - | 1   | CH1    | CH1         |
        - | 2   | CH2    | CH2         |
        - | 3   | Both   | CH1&CH2     |

  - Answer message (when a query is run)
    - param: Current coherence calculation mode
      - Format: NR1 (characters: 2) or string (to be set by SETup Mnemonic)
        - | Answer format | Description |
        - | NR1 | String |          |
        - |-----|--------|----------|
        - | 0   | OFF    | OFF      |
        - | 1   | CH1    | CH1      |
        - | 2   | CH2    | CH2      |
        - | 3   | BOTH   | CH1&CH2  |

  - Example setup
    - measure coherence CH1 Coherence mode: CH1

  - Remarks
    - If this is set to Off, a calculation process will be executed internally in CH1&CH2 (Both).

  - Related program code
    - Measure Auto mode
3.4 Individual program code description

- **Measure Delay Cycle**

  ?Measure Delay Cycle

  Setup: Measure Delay Cycle <param>
  
  Query: ?Measure Delay Cycle
          (Answer) MEASURE DELAY CYCLE <param>

  - General: Sets or inquires the delay cycles for measurement start

  - Parameter (when parameters are set)
    - param: Delay cycles
      Format: NR1
      Range: 0 to 9999 (cycles)

  - Answer message (when a query is run)
    - param: Current delay cycles
      Format: NR1
      Range: 0 to 9999
      Characters: 5

  - Example setup
    measure delay cycle 10               Delay cycles for measurement start: 10

  - Remarks
    Since the delay cycles indicate the number of cycles (= the reciprocal of the frequency to be analyzed), the delay time will vary according to the frequency to be analyzed. This setup will not be valid unless "Measure Delay Type" is set to 0 or Cycle.

  - Related program codes
    Measure Delay Sec, Measure Delay Type
3.4 Individual program code description

**Measure Delay Sec**

**?Measure Delay Sec**

Setup:  
**Measure Delay Sec <param>**

Query:  
**?Measure Delay Sec**

(Answer)  
**MEASURE DELAY SEC <param>**

- **General:** Sets or inquires the delay time for measurement start

- **Parameter (when parameters are set)**
  - **param:** Delay time for measurement start
    - **Format:** NR3
    - **Range:** 0 to 9999.0 (sec)
    - **Resolution:** 10 msec

- **Answer message (when a query is run)**
  - **param:** Current delay time
    - **Format:** NR3
    - **Range:** 0.00000E+00 to 9.99900E+03
    - **Characters:** 12

- **Example setup**
  
  ```
  measure delay sec 0.1
  ```

  Delay in measurement start: 0.1 secs

- **Remarks**
  
  This command will define the delay time that can be set regardless of the frequency to be analyzed.

  This setup will not be valid unless "Measure Delay Type" is set to 1 or Sec.

- **Related program codes**
  
  Measure Delay Cycle, Measure Delay Type
3.4 Individual program code description

- **Measure Delay Type**

  **?Measure Delay Type**

  Setup:  Measure Delay Type <param>
  
  Query:  ?Measure Delay Type
         (Answer) MEASURE DELAY TYPE <param>

  - General: Sets or inquires the setting type of delay in measurement start

  - Parameter (when parameters are set)
    - param: Setting type of delay in measurement start
    - Format: NR1 or string
    
    | NR1 | String | Description                  |
    |-----|--------|------------------------------|
    | 0   | Cycle  | Set by a number of times (cycles) |
    | 1   | Sec    | Set by time                  |

  - Answer message (when a query is run)
    - param: Current setting type of delay in measurement start
    - Format: NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

    | Answer format | Description                  |
    |---------------|------------------------------|
    | 0  CYCLE      | Set by a number of times (cycles) |
    | 1  SEC        | Set by time                  |

  - Example setup
    
    measure delay type sec  Sets delay in measurement start by means of time

  - Remarks
    
    Although the delay in measurement start can be set individually using "Measure Delay Cycle" and "Measure Delay Sec", it is the delay set by this program code that will actually be used.

  - Related program codes
    
    Measure Delay Cycle, Measure Delay Sec
3.4 Individual program code description

**Measure Harmonic**

**Setup:** Measure Harmonic <param>

**Query:** ?Measure Harmonic

(Answer) MEASURE HARMONIC <param>

- **General:** Sets or inquires the order of harmonic analysis

- **Parameter (when parameters are set)**
  - **param:** Order of harmonics to be analyzed
    - **Format:** NR1
    - **Range:** 1 to 10 (orders)
    - **Resolution:** 1

- **Answer message (when a query is run)**
  - **param:** Current order of harmonics to be analyzed
    - **Format:** NR1
    - **Range:** 1 to 10
    - **Characters:** 3

- **Example setup**

  measure harmonic 3  
  Analyzes the 3rd order harmonic content
3.4 Individual program code description

- **Measure Integration Cycle**

  ?Measure Integration Cycle

  **Setup:** Measure Integration Cycle <param>
  **Query:** ?Measure Integration Cycle
  (Answer) MEASURE INTEGRATION CYCLE <param>

  - **General:** Sets or inquires the cycles of manual integration

  - **Parameter (when parameters are set)**
    - param: Cycles of integration
      - Format: NR1
      - Range: 1 to 9999 (cycles)

  - **Answer message (when a query is run)**
    - param: Current cycles of integration
      - Format: NR1
      - Range: 1 to 9999
      - Characters: 5

  - **Example setup**
    - measure integration cycle 10  10 cycles of manual integration

  - **Remarks**
    - This setup will not be valid unless "Measure Integration Type" is set to 0 or Cycle.
      - The set cycles of integration will be run regardless of the cycles of the frequency to be analyzed.
      - Shown below is the approximate measuring time when the frequency to be analyzed is defined as \( f \) and the cycles of integration as \( n \).
        - \( f \leq \approx \text{approx 54 Hz:} \) \( n/f \)
        - \approx \text{approx 54 Hz}<f<3 \text{ kHz:} \ (18.2 \text{ to 54.6 ms}) \times n
        - \( 3 \text{ kHz } \leq f: \approx \text{approx 18.2 ms} \times n

  - **Related program codes**
    - Measure Integration Sec, Measure Integration Type
3.4 Individual program code description

■ Measure Integration Sec

?Measure Integration Sec

Setup: Measure Integration Sec <param>
Query: ?Measure Integration Sec
(Answer) MEASURE INTEGRATION SEC <param>

● General: Sets or inquires the integration time

● Parameter (when parameters are set)
  • param: Integration time
    Format: NR3
    Range: 0 to 9999.0 (secs)
    Resolution: 10 msecs

● Answer message (when a query is run)
  • param: Current integration time
    Format: NR3
    Range: 0.00000E+00 to 9.99900E+03
    Characters: 12

● Example setup
  measure integration sec 1  Integration time: 1 sec

● Remarks
  The integration time will be set regardless of the frequency to be analyzed.
  This setup will not be valid unless "Measure Integration Type" is set to 1 or sec.

● Related program codes
  Measure Integration Cycle, Measure Integration Type
3.4 Individual program code description

- **Measure Integration Type**

  **Measure Integration Type**

  *Setup:* Measure Integration Type <param>

  *Query:* ?Measure Integration Type

  (Answer) MEASURE INTEGRATION TYPE <param>

  - **General:** Sets or inquires the setting type of integration

  - **Parameter (when parameters are set):**
    - **param:** Setting type of integration
      - **Format:** NR1 or string
      
      | NR1 | String | Description                      |
      |-----|--------|----------------------------------|
      | 0   | Cycle  | Set by a number of times (cycles) |
      | 1   | Sec    | Set by time                      |

  - **Answer message (when a query is run):**
    - **param:** Current setting type of integration
      - **Format:** NR1 (characters: 2) or string (to be set by SETUP Mnemonic)

      | Answer format | Description                      |
      |---------------|----------------------------------|
      | 0 CYCLE       | Set by a number of times (cycles) |
      | 1 SEC         | Set by time                      |

  - **Example setup**
    
    measure integration type cycle  Sets integration by a number of times

  - **Remarks**
    
    Although integration can be set individually using "Measure Integration Cycle" and "Measure Integration Sec", it is the integration level set by this program code that will actually be used.

  - **Related program codes**
    
    Measure Integration Cycle, Measure Integration Sec
3.4 Individual program code description

**Measure Mode**

?Measure Mode

**Setup:**  Measure Mode <param>

**Query:**  ?Measure Mode  

(Answer)  MEASURE MODE <param>

- **General:**  Sets or inquires the measurement mode

- **Parameter (when parameters are set):**
  - **param:**  Measurement mode

  **Format:**  NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>CH1Ch2</td>
<td>CH1,CH2</td>
</tr>
<tr>
<td>1</td>
<td>CH1Osc</td>
<td>CH1,OSC</td>
</tr>
<tr>
<td>2</td>
<td>OscCh2</td>
<td>OSC,CH2</td>
</tr>
</tbody>
</table>

  **Omit:**  CH1,CH2

- **Answer message (when a query is run):**
  - **param:**  Current measurement mode

  **Format:**  NR1 (characters: 2) or string (to be set by **SE**tup **Mn**emonic)

  **Answer format**

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>CH1CH2</td>
<td>CH1,CH2</td>
</tr>
<tr>
<td>1</td>
<td>CH1OSC</td>
<td>CH1,OSC</td>
</tr>
<tr>
<td>2</td>
<td>OSCCH2</td>
<td>OSC,CH2</td>
</tr>
</tbody>
</table>

- **Example setup**

  ```
  measure mode CH1CH2  
  Measurement mode: CH1,CH2
  ```
3.4 Individual program code description

### Measure Repeat

**?Measure Repeat**

**Setup:** Measure Repeat <param>

**Query:** ?Measure Repeat  
(Answer) MEASURE REPEAT <param>

- **General:** Sets or inquires the measurement (Repeat) mode

- **Parameter (when parameters are set)**
  - **param:** Setting of the measure repeat mode
    
    **Format:** NR1 or string

    | NR1 | String | Description |
    |-----|--------|-------------|
    | 0   | OFF   | OFF         |
    | 1   | ON    | ON          |

- **Answer message (when a query is run)**
  - **param:** Current measure repeat mode
    
    **Format:** NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

    | Answer format | Description |
    |--------------|-------------|
    | NR1 String   |             |
    | 0            | OFF         |
    | 1            | ON          |

- **Example setup**
  
  ```
  measure repeat 0
  ```

  Turns OFF the measure repeat mode

- **Remarks**

  This command corresponds to the operation that is executed by the **REPEAT** key on the front panel.
3.4 Individual program code description

- **OSCillator Amplitude**

  **?OSCillator Amplitude**

  **Setup:**  
  OSCillator Amplitude <param>

  **Query:**  
  ?OSCillator Amplitude
  (Answer) OSCILLATOR AMPLITUDE <param>

  - **General:** Sets or inquires the output amplitude of the oscillator

  - **Parameter (when parameters are set)**
    
    - **param:** Output amplitude (when output open)
      
      - Format: NR3
      - Range: 0 to 10 (Vpeak)
      - Resolution: Up to 3 digits, 10 μV

  - **Answer message (when a query is run)**
    
    - **param:** Current output amplitude (when output open)
      
      - Format: NR3
      - Range: 0.00E-00 to 10.0E+00
      - Characters: 9

  - **Example setup**
    
    oscillator amplitude 5.00  Sets the amplitude to 5 V

  - **Remarks**
    
    Even if the output amplitude is modified with this program code, the actual output amplitude will not be changed unless the oscillator output is set to ON in the "OSCillator mode".

  - **Related program code**
    
    OSCillator mode
3.4 Individual program code description

- Oscillator Frequency

?Oscillator Frequency

Setup: Oscillator Frequency <param>
Query: ?Oscillator Frequency
(Answer) OSCILLATOR FREQUENCY <param>

- General: Sets or inquires the oscillator frequency

- Parameter (when parameters are set)
  - param: Oscillator frequency
    Format: NR3
    Range: 0.1E-3 to 15E+6(Hz)
    Resolution: 0.1E - 3 (0.1 mHz)

- Answer message (when a query is run)
  - param: Current oscillator frequency
    Format: NR3
    Range: 0.1E-03 to 15.000000000E+06
    Characters: 17

- Example setup
  oscillator frequency 1.0e6               Sets the oscillator frequency to 1 MHz
3.4 Individual program code description

### Oscillator mode

#### ?Oscillator mode

**Setup:** Oscillator mode <param1>, <param2>, <param3>

**Query:** ?Oscillator mode

(Answer) Oscillator mode <param1>, <param2>, <param3>

- **General:** Sets or inquires the voltage change mode

- **Parameter (when parameters are set):**
  - **param1:** Oscillator ON/OFF
    - **Format:** NR1 or string
    - **Description:**
      | NR1 | String | Description          |
      |-----|--------|----------------------|
      | 0   | OFF    | AC/DC OFF            |
      | 1   | Acoff  | Only AC is OFF       |
      | 2   | ON     | AC/DC ON             |
      | Omit|        | None changes         |

  - **param2:** Voltage change mode of oscillator (SLOW ON/OFF setup)
    - **Format:** NR1 or string
    - **Description:**
      | NR1 | String | Description               |
      |-----|--------|---------------------------|
      | 0   | Quick  | Output changes just after ON/OFF |
      | 1   | Slow   | SLOW ON/OFF mode          |
      | Omit|        | None changes               |

  - **param3:** Oscillator stop mode
    - **Format:** NR1 or string
    - **Description:**
      | NR1 | String | Description             |
      |-----|--------|-------------------------|
      | 0   | Zero   | Stops at 0 V            |
      | 1   | Hold   | Stops in current phase  |
      | 2   | Phase  | Stops in specified phase*1 |
      | Omit|        | None changes            |

*1: The stop phase can be set by "oscillator phase".
• Answer message (when a query is run)
  • param1: Current ON/OFF status of oscillator
    Format: NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>OFF</td>
<td>AC/DC OFF</td>
</tr>
<tr>
<td>1</td>
<td>ACOFF</td>
<td>Only AC is OFF</td>
</tr>
<tr>
<td>2</td>
<td>ON</td>
<td>AC/DC ON</td>
</tr>
<tr>
<td>3</td>
<td>TRANSITION</td>
<td>In transition</td>
</tr>
</tbody>
</table>

• param2: Voltage change mode of current oscillator
  Format: NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>QUICK</td>
<td>Output changes just after ON/OFF</td>
</tr>
<tr>
<td>1</td>
<td>SLOW</td>
<td>SLOW ON/OFF mode</td>
</tr>
</tbody>
</table>

• param3: Current oscillator stop mode
  Format: NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>ZERO</td>
<td>Stops at 0 V</td>
</tr>
<tr>
<td>1</td>
<td>HOLD</td>
<td>Stops in current phase</td>
</tr>
<tr>
<td>2</td>
<td>PHASE</td>
<td>Stops in specified phase</td>
</tr>
</tbody>
</table>

• Example setup
  oscillator mode off,0,0  Turns AC/DC OFF at once. The stop phase is 0 V.

  ?oscillator mode
  (Answer) OFF, QUICK, ZERO

• Related program code
  OScillator Phase
3.4 Individual program code description

- **OSCillator OFFSet**

  ?OSCillator OFFSet

  Setup:  OSCillator OFFSet <param>
  Query:  ?OSCillator OFFSet
          (Answer)  OSCILLATOR OFFSET <param>

  - General: Sets or inquires the DC bias

  - Parameter (when parameters are set)
    - param: DC bias (when output open)
      Format: NR3
      Range: -10 to +10 (V)
      Resolution: 10E-3 (10 mV)

  - Answer message (when a query is run)
    - param: Current DC bias (when output open)
      Format: NR3
      Range: -10.00E+00 to +10.00E+00
      Characters: 10

  - Example setup
    oscillator offset 1.0   DC bias: 1.0 V

  - Remarks
    Even if the output bias is modified by this program code, the actual output bias will not be changed unless the oscillator output is set to ON in the "OSCillator mode".

  - Related program code
    OSCillator mode
3.4 Individual program code description

- **OScillator Phase**

  ?OScillator Phase

  **Setup:**  `OScillator Phase <param>`
  **Query:**  `?OScillator Phase`  
  (Answer)  `OSCILLATOR PHASE <param>`

  - **General:** Sets or inquires the start/stop phase of the oscillator

  - **Parameter (when parameters are set)**
    - **param:** Start/stop phase
      - **Format:**  `NR1`
      - **Range:**  `0 to 359 (deg)`
      - **Resolution:**  `1 (deg)`

  - **Answer message (when a query is run)**
    - **param:** Current start/stop phase
      - **Format:**  `NR1`
      - **Range:**  `0 to 359`
      - **Characters:**  `4`

  - **Example setup**
    
    oscillator phase 90  
    
    Start/stop phase: 90 deg  
    
    Stops at a peak on the positive side of sine wave

  - **Remarks**
    
    If this command has been received while the oscillator is ON (either AC or DC or both), an operation in sequence of SLOW OFF → SLOW ON → output ON will result, so that the oscillator output will be turned OFF temporarily. When the oscillator is in process of SLOW ON (i.e., while the amplitude is increasing), an operation of SLOW OFF → SLOW ON will start in a new phase. Whether or not the oscillator amplitude is changing due to SLOW ON or SLOW OFF, can be checked in the “?OScillator mode”.

  - **Related program code**
    
    OScillator mode
3.4 Individual program code description

- Oscillator Waveform

?Oscillator Waveform

Setup: Oscillator Waveform <param>
Query: ?Oscillator Waveform
(Answer) OSCILLATOR WAVEFORM <param>

- General: Sets or inquires the oscillator waveform

- Parameter (when parameters are set)
  - param: Selected waveform
    Format: NR1 or string
    \[
    \begin{array}{|c|c|c|}
    \hline
    \text{NR1} & \text{String} & \text{Description} \\
    \hline
    0 & \text{S}i\text{n}\text{e} & \text{Sine wave} \\
    1 & \text{S}q\text{u}\text{a}\text{r}\text{e} & \text{Square wave} \\
    2 & \text{Tri}\text{a}\text{ngle} & \text{Triangular wave} \\
    \hline
    \end{array}
    \]

- Answer message (when a query is run)
  - param: Current waveform
    Format: NR1 (characters: 2) or string (to be set by SETup Mnemonic)
    \[
    \begin{array}{|c|c|c|}
    \hline
    \text{Answer format} & \text{String} & \text{Description} \\
    \hline
    0 & \text{S}\text{I}\text{N}\text{E} & \text{Sine wave} \\
    1 & \text{S}\text{Q}\text{U}\text{A}\text{R}E & \text{Square wave} \\
    2 & \text{T}\text{R}\text{I}\text{A}\text{N}\text{G}E & \text{Triangular wave} \\
    \hline
    \end{array}
    \]

- Example setup
  oscillator waveform 0 Sets the oscillator waveform to sine wave
**SEtup Buzzer**

**?SEtup Buzzer**

Setup:  SEtup Buzzer <param>

Query:  ?SEtup Buzzer

(Answer)  SETUP BUZZER <param>

- **General:** Sets the buzzer to ON/OFF

- **Parameter (when parameters are set)**
  - **param:** Buzzer ON/OFF

  **Format:** NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>OFF</td>
<td>Buzzer OFF</td>
</tr>
<tr>
<td>1</td>
<td>ON</td>
<td>Buzzer ON</td>
</tr>
</tbody>
</table>

- **Answer message (when a query is run)**
  - **param:** Current buzzer ON/OFF setup

  **Format:** NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>OFF</td>
</tr>
<tr>
<td>1</td>
<td>Buzzer ON</td>
</tr>
</tbody>
</table>

- **Example setup**

  setup buzzer ON  Sets the buzzer to ON

- **Remarks**

  This command will determine whether or not to sound a buzzer when an overload input has been detected or an error has occurred. The click that is heard when hitting a key is not affected by this setup.
### SEtpup Calibration

**Setup:** SEtpup Calibration

- **General:** Performs a calibration

- **Example setup**
  - setup calibration Performs a calibration

- **Remarks**
  FRA5097 error compensation (calibration) is conducted. End of calibration is indicated by Bit 1 of status byte. See Table 2-1 Status Byte.
3.4 Individual program code description

### Setup Date

**Setup**: `SEtup Date <param1>, <param2>, <param3>`

**Query**: `?SEtup Date`

(Answer) `SETUP DATE <param1>, <param2>, <param3>`

- **General**: Sets or inquires the date

- **Parameter (when parameters are set)**
  - `param1`: Year
    - Format: NR1
    - Range: 1970 to 2069
    - omit: None changes
  - `param2`: Month
    - Format: NR1
    - Range: 1 to 12
    - omit: None changes
  - `param3`: Day
    - Format: NR1
    - Range: 1 to 31
    - omit: None changes

- **Answer message (when a query is run)**
  - `param1`: Year
    - Format: NR1, 5 characters
    - Range: 1970 to 2069
  - `param2`: Month
    - Format: NR1, 3 characters
    - Range: 1 to 12
  - `param3`: Day
    - Format: NR1, 3 characters
    - Range: 1 to 31

- **Example setup**
  - `setup date 2001, 7, 14` Sets the internal calendar to July 14, 2001

- **Related program code**
  - `SEtup Time`
3.4 Individual program code description

**SEtup Header**

??SEtup Header

Setup: **SEtup Header** <param>

Query: ??SEtup Header

(Answer) **SETUP HEADER** <param>

- **General:** Sets or inquires the header ON/OFF status

- **Parameter (when parameters are set)**
  - **param:** Setup of header ON/OFF

  **Format:** NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td><strong>OFF</strong></td>
<td>Header OFF</td>
</tr>
<tr>
<td>1</td>
<td><strong>ON</strong></td>
<td>Header ON</td>
</tr>
</tbody>
</table>

- **Answer message (when a query is run)**
  - **param:** Current setup of header ON/OFF

  **Format:** NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 OFF</td>
<td>Header OFF</td>
</tr>
<tr>
<td>1 ON</td>
<td>Header ON</td>
</tr>
</tbody>
</table>

- **Example setup**

  setup header ON    
  
  Sets the header to ON

- **Remarks**

  This command will set whether or not to prefix a header to an answer message to a query message (a program code prefixed with "?").

  The header will be set to OFF when a DCL or a SDC has been received or by power on.
3.4 Individual program code description

**SE**tup Initialize

Setup: **SE**tup Initialize

- **General**: Sets the initialization

- **Example setup**
  
  setup initialize Initializes

- **Remarks**
  
  If this command is executed, the settings on the FRA5097 will be initialized, except that the following ones will remain.
  
  - GPIB address
  - GPIB delimiter
  - Contents of mass memory and permanent memory
  - External control selection (GPIB/USB)

  For details on initialization, refer to the "FRA5097 INSTRUCTION MANUAL".
3.4 Individual program code description

**SETup Mnemonic**

`?SETup Mnemonic`

Setup:  `SETup Mnemonic <param>`
Query:  `?SETup Mnemonic`

(Answer)  `SETUP MNEMONIC <param>`

- **General:** Sets or inquires the numeric format/mnemonic format for an answer string

- **Parameter (when parameters are set)**
  - **param:** Mnemonic ON/OFF setup
  
  formatting

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Off</td>
<td>Numeric format</td>
</tr>
<tr>
<td>1</td>
<td>On</td>
<td>Mnemonic format</td>
</tr>
</tbody>
</table>

- **Answer message (when a query is run)**
  - **param:** Current mnemonic ON/OFF setup
  
  formatting

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Numeric format</td>
</tr>
<tr>
<td>–</td>
<td>Mnemonic format</td>
</tr>
</tbody>
</table>

- **Example setup**

  `setup mnemonic ON`  
  Sets the answer format to mnemonic

- **Remarks**

  This command will determine whether to use the NR1 format (Mnemonic OFF) or the mnemonic format (Mnemonic ON) for the NR1- or string-format answer messages to queries.
3.4 Individual program code description

**SEtup Time**

*SE*etup Time

Setup:  **SEtup Time** <param1>, <param2>, <param3>

Query:  **?SE*etup Time**

(Answer)  **SETUP TIME** <param1>, <param2>, <param3>

- **General:** Sets or inquires the time

- **Parameter (when parameters are set):**
  - param1:  Time
    Format:  NR1  Range:  0 to 23 (24-hour system)
    Omit:  None changes
  - param2:  Minute
    Format:  NR1  Range:  0 to 59
    Omit:  None changes
  - param3:  Second
    Format:  NR1  Range:  0 to 59
    Omit:  None changes

- **Answer message (when a query is run):**
  - param1:  Time
    Format:  NR1, 3 characters  Range:  0 to 23
  - param2:  Minute
    Format:  NR1, 3 characters  Range:  0 to 59
  - param3:  Second
    Format:  NR1, 3 characters  Range:  0 to 59

- **Example setup**
  setup time 14, 0, 0  Sets the internal clock to 14:00

- **Related program code**
  SEtup Date
3.4 Individual program code description

**SW**eeP **MA**nual

**?SWeeP **MA**nual**

Setup:  **SWeeP MA**nual <param>

Query:  **?SWeeP MA**nual

(Answer)  **SWEEP MANUAL** <param>

- **General**: Sets or inquires the manual/automatic sweep

- **Parameter (when parameters are set)**
  - **param**: Sweep type
    
    Format: NR1 or string

    | NR1 | String | Description       |
    |-----|--------|-------------------|
    | 0   | OFF    | Automatic sweep   |
    | 1   | ON     | Manual sweep      |

- **Answer message (when a query is run)**
  - **param**: Current sweep type
    
    Format: NR1 (characters: 2) or string (to be set by SE*up Mnemonic)

    | Answer format | String | Description       |
    |---------------|--------|-------------------|
    | 0             | OFF    | Automatic sweep   |
    | 1             | ON     | Manual sweep      |

- **Example setup**
  
  `sweep manual 0`  
  Sets the automatic sweep mode

- **Related program code**
  
  SWeeP ME*asure
3.4 Individual program code description

■ SWeep MEasure

?SWeep MEasure

Setup:  SWeep MEasure <param>
Query:  ?SWeep MEasure
  (Answer)  SWEEP MEASURE <param>

- General: Controls or inquires a sweep measurement

- Parameter (when parameters are set)
  - param: Control of a sweep measurement
    Format:  NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Stop</td>
<td>Interrupts a sweep</td>
</tr>
<tr>
<td>1</td>
<td>Hold</td>
<td>Holds a sweep; or starts Single/Repeat measurement</td>
</tr>
<tr>
<td>2</td>
<td>Up</td>
<td>Starts Up sweep</td>
</tr>
<tr>
<td>3</td>
<td>Down</td>
<td>Starts Down sweep</td>
</tr>
</tbody>
</table>

- Answer message (when a query is run)
  - param: Current status of sweep measurement
    Format:  NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 STOP</td>
<td>Sweep stopped status</td>
</tr>
<tr>
<td>1 HOLD</td>
<td>Sweep held status or Single/Repeat measurement in progress</td>
</tr>
<tr>
<td>2 UP</td>
<td>Up sweep in progress</td>
</tr>
<tr>
<td>3 DOWN</td>
<td>Down sweep in progress</td>
</tr>
<tr>
<td>-1 CALIBRATION</td>
<td>Calibration in progress</td>
</tr>
</tbody>
</table>

- Example setup
  sweep measure up  Starts an Up sweep measurement

- Remarks
  The action that results from the program code "SWeep MEasure Hold", will vary depending on the status of the FRA5097 the moment it has received that program code.

  - When an Up or Down sweep is in progress: Measurement will be held.
  - When a measurement is stopped: Single or Repeat measurement will start.

- Related program codes
  SWeep MANual, SWeep range, SWeep RESolution ***
### SWeep range

#### ?SWeep range

Setup:  SWeep range <param1>, <param2>
Query:  ?SWeep range  

(Answer)  SWEEP RANGE <param1>, <param2>

- **General:** Sets or inquires the sweep frequency range

- **Parameter:** (when parameters are set)
  - param1: Minimum sweep frequency (Hz)
  - param2: Maximum sweep frequency (Hz)
  - Format: NR3
  - Range: 0.1E-3 to 15E+6 (Hz)
  - Resolution: 0.1E-3 (Hz)
  - omit: None changes

- **Answer message:** (when a query is run)
  - param1: Current minimum sweep frequency (Hz)
  - param2: Current maximum sweep frequency (Hz)
  - Format: NR3
  - Range: 0.1E-03 to 15.000000000E+06
  - Characters: 17

- **Example setup**
  - sweep 1, 1e6  
  - sweep range ,,2.2e6  
  - sw 1e3,  

- **Related program code**
  - SWeep REsolution ***
### SWeep REsolution log sweep

**?SWeep REsolution log sweep**

**Setup:**  
`SWEEP RESOLUTION LOG SWEEP <param>`

**Query:**  
`?SWEEP RESOLUTION LOG SWEEP`

**(Answer)**  
`SWEEP RESOLUTION LOG SWEEP <param>`

- **General:** Sets or inquires the resolution (steps/sweep) of a log sweep

- **Parameter (when parameters are set)**
  - `param:` Resolution of a log sweep
    - Format: NR1
    - Range: 3 to 20000 (steps/sweep)

- **Answer message (when a query is run)**
  - `param:` Current resolution of a log sweep
    - Format: NR1
    - Range: 3 to 20000
    - Characters: 6

- **Example setup**
  - `sweep resolution 100`  
  
  (Log) sweep resolution: 100 steps/sweep

- **Remarks**

  If the sweep resolution, the minimum sweep frequency and the maximum sweep frequency are set to `n` (steps/sweep), `f_L` (Hz), and `f_H` (Hz), respectively, the frequency point where a sweep measurement is carried out, will be determined by the following formula.

  ```latex
  f_L \left(\frac{f_H}{f_L}\right)^{0/n} = f_L  
  f_L \left(\frac{f_H}{f_L}\right)^{1/n}  
  \vdots  
  f_L \left(\frac{f_H}{f_L}\right)^{n-1/n}  
  f_L \left(\frac{f_H}{f_L}\right)^{n/n} = f_H
  ```

  The number of measurement data that is obtained by a sweep will be the set sweep resolution (steps/sweep) plus one.

  This setup will be valid when "SWEEP RESOLUTION MODE" is set to 0 or LOGSweep.

- **Related program code**
  - `SWEEP RESOLUTION MODE`
3.4 Individual program code description

**SWeep REsolution log D ecade**

**?SWeep REsolution log D ecade**

Setup:  **SWeep REsolution log D ecade <param>**

Query:  **?SWeep REsolution log D ecade**

   (Answer)  **SWEEP RESOLUTION LOG DECADE <param>**

- **General:** Sets or inquires the per-digit resolution of a log sweep (steps/decade)

- **Parameter (when parameters are set)**
  - param:  Per-digit resolution of a log sweep
    - Format:  NR1
    - Range:  1 to 20000 (steps/decade)

- **Answer message (when a query is run)**
  - param:  Current per-digit resolution of a log sweep
    - Format:  NR1
    - Range:  1 to 20000
    - Characters:  6

- **Example setup**
  
  sweep resolution log decade 10  
  
  (Log) sweep resolution: 10steps/decade

- **Remarks**
  
  This setup will be valid when "SWeep REsolution Mode" is set to 1 or LOGDecade.

- **Related program code**
  
  SWeep REsolution Mode
3.4 Individual program code description

- **SWeep REsolution LIN sweep**

  
  **?SWeep REsolution LIN sweep**

  Setup:  `SWEEP REsolution LIN sweep <param>`

  Query:  `?SWEEP REsolution LIN sweep`

  (Answer)  `SWEEP RESOLUTION LIN SWEEP <param>`

  - **General:** Sets or inquires the resolution of a linear sweep (steps/sweep)

  - **Parameter (when parameters are set)**
    - `param:` Resolution of a linear sweep
      - Format: `NR1`
      - Range: 3 to 20000 (steps/sweep)

  - **Answer message (when a query is run)**
    - `param:` Current resolution of a linear sweep
      - Format: `NR1`
      - Range: 3 to 20000
      - Characters: 6

  - **Example setup**
    
    `sweep resolution lin sweep 100`  
    Linear sweep resolution: 100 steps/sweep

  - **Remarks**
    
    This will set the resolution of a sweep that is performed at equal intervals between the maximum and minimum sweep frequencies. The number of measurement data will be the value that was set by this program code plus one.

    This setup will be valid when "SWeep REsolution Mode" is set to 2 or LINSwEEP.

  - **Related program code**
    
    `SWEEP REsolution Mode`
3.4 Individual program code description

- **SW**eeP **RE**solution **LIn** Hz

  ?**SW**eeP **RE**solution **LIn** Hz

Setup: **SW**eeP **RE**solution **LIn** Hz <param>
Query: ?**SW**eeP **RE**solution **LIn** Hz
  (Answer) **SWEEP RESOLUTION LIN HZ** <param>

  - **General:** Sets or inquires the resolution of a linear sweep (frequency)

  - **Parameter (when parameters are set)**
    - **param:** Resolution of a linear sweep
      Format: NR3
      Range: 0.1E-3 to 15E+6 (Hz)

  - **Answer message (when a query is run)**
    - **param:** Current resolution of a linear sweep
      Format: NR3
      Range: 0.1E-03 to 15.000000000E+6
      Characters: 17

  - **Example setup**
    sweep resolution lin hz 1e3     Sets a sweep measurement per 1 kHz

  - **Remarks**
    This setup will be valid when "SWEEP RESOLUTION Mode" is set to 3 or LINHz.

  - **Related program code**
    SWEEP RESOLUTION Mode
### SWeep RESolution Mode

**?SWeep RESolution Mode**

**Setup:** `SWeep RESolution Mode <param>`

**Query:** `?SWeep RESolution Mode`  
  (Answer) `SWEEP RESOLUTION MODE <param>`

- General: Sets or inquires the type of sweep resolution

- Parameter (when parameters are set)
  - **param:** Type of sweep resolution
    - Format: NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LOGSweep</td>
<td>Log sweep. Setup by SWEEP RESOLUTION log sweep is valid</td>
</tr>
<tr>
<td>1</td>
<td>LOGDecade</td>
<td>Log sweep. Setup by SWEEP RESOLUTION log Decade is valid</td>
</tr>
<tr>
<td>2</td>
<td>LINSweep</td>
<td>Linear sweep. Setup by SWEEP RESOLUTION LIn sweep is valid</td>
</tr>
<tr>
<td>3</td>
<td>LINHz</td>
<td>Linear sweep. Setup by SWEEP RESOLUTION LIn Hz is valid</td>
</tr>
</tbody>
</table>

- Answer message (when a query is run)
  - **param:** Current status of a sweep measurement
    - Format: NR1 (characters: 2) or string (to be set by SETup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LOGSweep. Setup by SWEEP RESOLUTION log sweep is valid</td>
</tr>
<tr>
<td>1</td>
<td>LOGDecade. Setup by SWEEP RESOLUTION log Decade is valid</td>
</tr>
<tr>
<td>2</td>
<td>LINSweep. Setup by SWEEP RESOLUTION LIn sweep is valid</td>
</tr>
<tr>
<td>3</td>
<td>LINHz. Setup by SWEEP RESOLUTION LIn Hz is valid</td>
</tr>
</tbody>
</table>

- Example setup
  - `sweep resolution mode 0` Log sweep. Enables the value set by SWEEP RESOLUTION log sweep
Remarks

The following four types of settings are available for a sweep resolution:

- SWEEP RESOLUTION log sweep (Log sweep. To be set by the number of steps in a sweep range)
- SWEEP RESOLUTION log Decade (Log sweep. To be set by the number of steps per digit of frequency)
- SWEEP RESOLUTION Lin sweep (Linear sweep. To be set by the number of steps in a sweep range)
- SWEEP RESOLUTION Lin Hz (Linear sweep. To be set by the variation range of a frequency)

Individual settings can be made by the corresponding program codes, and the set values will be stored internally in the FRA5097. When a sweep measurement is actually carried out, the sweep resolution type set by the program code "SWEEP RESOLUTION Mode" will be selected. The set values for the remaining three sweep resolution types will be stored as is.

Related program codes

SWEEP RESOLUTION log sweep, SWEEP RESOLUTION log Decade,
SWEEP RESOLUTION Lin sweep, SWEEP RESOLUTION Lin Hz, SWEEP range
3.4 Individual program code description

- **SWeeP Slow mode**

  *?SWeeP Slow mode*

  **Setup:**  
  `SWeeP Slow mode <param>`

  **Query:**  
  `?SWeeP Slow mode`

  `(Answer)`  
  `SWEEP SLOW MODE <param>`

  - **General:** Sets or inquires the operation mode of a low-speed & high-density sweep

  - **Parameter (when parameters are set)**
    - **param:** Operation mode of a low-speed, high density sweep (SlowSweep)

      **Format:** NR1 or string

      | NR1 | String | Description     |
      |-----|--------|-----------------|
      | 0   | OFF   | Function OFF    |
      | 1   | Auto  | Auto mode      |
      | 2   | Manual| Manual mode    |

  - **Answer message (when a query is run)**

    - **param:** Operation mode of the current low-speed, high density sweep

      **Format:** NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

      **Answer format**

      | NR1 | String | Description     |
      |-----|--------|-----------------|
      | 0   | OFF   | Function OFF    |
      | 1   | AUTO  | Auto mode      |
      | 2   | MANUAL| Manual mode    |

  - **Example setup**

    ```
    sweep slow mode 1
    ```

    Sets the low-speed & high-density sweep of AUTO mode

  - **Related program codes**

    `SWeeP Slow Target, SWeeP Slow Variation ***`
### SWeep Slow Target

**?SWeep Slow Target**

**Setup:**  \texttt{SWeep Slow Target <param>}

**Query:**  \texttt{?SWeep Slow Target}

(Answer)  \texttt{SWEEP SLOW TARGET <param>}

- **General:** Sets or inquires the reference channel of a low-speed, high-density sweep

- **Parameter (when parameters are set)**
  - **param:** Reference channel of a low-speed, high density sweep

  **Format:** NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CH1</td>
<td>Reference channel: CH1</td>
</tr>
<tr>
<td>2</td>
<td>CH2</td>
<td>Reference channel: CH2</td>
</tr>
</tbody>
</table>

- **Answer message (when a query is run)**
  - **param:** Reference channel of the current low-speed, high density sweep

  **Format:** NR1 (characters: 2) or string (to be set by SEtup Mnemonic)

<table>
<thead>
<tr>
<th>Answer format</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NR1</td>
<td>CH1</td>
<td>Reference channel: CH1</td>
</tr>
<tr>
<td>2</td>
<td>CH2</td>
<td>Reference channel: CH2</td>
</tr>
</tbody>
</table>

- **Example setup**

  ```
  sweep slow target ch1
  ```

  Refers to CH1 level and starts a low-speed & high-density sweep

- **Related program codes**

  SWeep Slow mode, SWeep Slow Variation ***
3.4 Individual program code description

**SW**eep **S**low **V**ariation **L**ogr

??SW**eep **S**low **V**ariation **L**ogr

Setup:  **SW**eep **S**low **V**ariation **L**ogr <param>

Query:  ??SW**eep **S**low **V**ariation **L**ogr

(Answer)  **SWEEP SLOW VARIATION LOGR** <param>

- **General:** Sets or inquires the target variation for a low-speed & high-density sweep (measuring gain: dB)

- **Parameter (when parameters are set)**
  - **param:** Target variation (measuring gain: dB)
    - Format: NR3
    - Range: 0 to 1000 (dB)

- **Answer message (when a query is run)**
  - **param:** Current target variation (measuring gain: dB)
    - Format: NR3
    - Range: 0.00E+00 to 1000E+00 (dB)
    - Characters: 9

- **Example setup**
  
  `sweep slow variation logr 0.1`  Starts a low-speed & high-density sweep when the measuring gain has changed by 0.1 dB or more than the last value measured

- **Remarks**

  This setup will be valid when "SWEEP Slow Variation Mode" is set to 0 or Logr.

- **Related program codes**

  SWEEP Slow mode, SWEEP Slow Target, SWEEP Slow Variation Mode
### 3.4 Individual program code description

#### SWeep Slow Variation R

**Setup:**  
`SWeep Slow Variation R <param>`

**Query:**  
`?SWeep Slow Variation R`  
*(Answer) SWEEP SLOW VARIATION R <param>*

- **General:** Sets or inquires the target variation (measuring voltage: Vrms) for a low-speed & high-density sweep

- **Parameter (when parameters are set)**
  - `param:` Target variation (measuring voltage: Vrms)
    - Format: NR3
    - Range: 0 to 1.0E+9 (Vrms)

- **Answer message (when a query is run)**
  - `param:` Current target variation (measuring voltage: Vrms)
    - Format: NR3
    - Range: 0.00E+00 to 1.00E+09 (Vrms)
    - Characters: 9

- **Example setup**
  
  ```
  sweep slow variation r 1.5
  ```

  Starts a low-speed & high-density sweep when the measuring voltage of the reference channel has changed by 1.5 Vrms or more than the last value measured

- **Remarks**
  
  This setup will be valid when "SWeep Slow Variation Mode" is set to 1 or R.

- **Related program codes**
  
  SWeep Slow mode, SWeep Slow Target, SWeep Slow Variation Mode
### SWeep Slow Variation Theta

**Query:** ?SWeep Slow Variation Theta

(Answer) SWEEP SLOW VARIATION THETA <param>

- **General:** Sets or inquires the target variation (phase: deg) for a low-speed & high-density sweep

- **Parameter (when parameters are set)**
  - **param:** Target variation (phase: deg)
    - Format: NR3
    - Range: 0 to 180 (deg)

- **Answer message (when a query is run)**
  - **param:** Current target variation (phase: deg)
    - Format: NR3
    - Range: 0.00E+00 to 180E+00
    - Characters: 9

- **Example setup**
  
  ```
  sweep slow variation theta 0.5
  ```

  Starts a low-speed & high-density sweep when the phase of the reference channel has changed by 0.5 deg or more than the last value measured

- **Remarks**
  This setup will be valid when "SWeep Slow Variation Mode" is set to 2 or Theta.

- **Related program codes**
  SWeep Slow mode, SWeep Slow Target, SWeep Slow Variation Mode
3.4 Individual program code description

■ SWeeep Slow Variation A

?SWeeep Slow Variation A

Setup:  SWeeep Slow Variation A <param>
Query:  ?SWeeep Slow Variation A
        (Answer)  SWEEP SLOW VARIATION A <param>

• General:  Sets or inquires the target variation (real voltage: Vrms) for a low-speed, high density sweep

• Parameter (when parameters are set)
  • param:  Target variation (real voltage: Vrms)
        Format:  NR3
        Range:  0.00E+00 to 1.00E+9 (Vrms)

• Answer message (when a query is run)
  • param:  Current target variation (real voltage: Vrms)
        Format:  NR3
        Range:  0.00E+00 to 1.00E+09 (Vrms)
        Characters:  9

• Example setup
  sweep slow variation a 5  Starts a low-speed & high-density sweep when the real voltage of the reference channel has changed by 5 Vrms or more than the last value measured

• Remarks
  This setup will be valid when "SWeep Slow Variation Mode" is set to 3 or A.

• Related program codes
  SWeep Slow mode, SWeep Slow Target, SWeep Slow Variation Mode
3.4 Individual program code description

■ SWeeP Slow Variation B

?SWeeP Slow Variation B

Setup:  SWeeP Slow Variation B <param>
Query:  ?SWeeP Slow Variation B
        (Answer)  SWEEP SLOW VARIATION B <param>

- General: Sets or inquires the target variation (imaginary voltage: Vrms) for a low-speed & high-density sweep

- Parameter (when parameters are set)
  - param: Target variation (imaginary voltage: Vrms)
    Format: NR3
    Range: 0.00E+00 to 1.00E+9 (Vrms)

- Answer message (when a query is run)
  - param: Current target variation (imaginary voltage: Vrms)
    Format: NR3
    Range: 0.00E+00 to 1.00E+09 (Vrms)
    Characters: 9

- Example setup
  sweep slow variation b 1.2  Starts a low-speed & high-density sweep when the imaginary voltage of the reference has changed by 1.2 Vrms or more than the last value measured

- Remarks
  This setup will be valid when "SWeeP Slow Variation Mode" is set to 4 or B.

- Related program codes
  SWeeP Slow mode, SWeeP Slow Target, SWeeP Slow Variation Mode
### SWeep Slow Variation Mode

**?SWeep Slow Variation Mode**

**Setup:**  
**SWeep Slow Variation Mode** <param>

**Query:**  
?SWeep Slow Variation Mode

(Answer)  
SWEEP SLOW VARIATION MODE <param>

- **General:** Sets or inquires the target variation type for a low-speed & high-density sweep

- **Parameter (when parameters are set)**
  - **param:** Target variation type for a low-speed & high-density sweep (SlowSweep)

**Format:** NR1 or string

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Logr</td>
<td>Setup by SWEEP Slow Variation Logr is valid</td>
</tr>
<tr>
<td>1</td>
<td>R</td>
<td>Setup by SWEEP Slow Variation R is valid</td>
</tr>
<tr>
<td>2</td>
<td>Theta</td>
<td>Setup by SWEEP Slow Variation Theta is valid</td>
</tr>
<tr>
<td>3</td>
<td>A</td>
<td>Setup by SWEEP Slow Variation A is valid</td>
</tr>
<tr>
<td>4</td>
<td>B</td>
<td>Setup by SWEEP Slow Variation B is valid</td>
</tr>
</tbody>
</table>

- **Answer message (when a query is run)**

  - **param:** Target variation type for the current low-speed & high-density sweep

**Format:**  
NR1 (characters: 2) or string (to be set by SETup Mnemonic)

**Answer format**

<table>
<thead>
<tr>
<th>NR1</th>
<th>String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LOGR</td>
<td>Setup by SWEEP Slow Variation Logr is valid</td>
</tr>
<tr>
<td>1</td>
<td>R</td>
<td>Setup by SWEEP Slow Variation R is valid</td>
</tr>
<tr>
<td>2</td>
<td>THETA</td>
<td>Setup by SWEEP Slow Variation THETA is valid</td>
</tr>
<tr>
<td>3</td>
<td>A</td>
<td>Setup by SWEEP Slow Variation A is valid</td>
</tr>
<tr>
<td>4</td>
<td>B</td>
<td>Setup by SWEEP Slow Variation B is valid</td>
</tr>
</tbody>
</table>

- **Example setup**

  sweep slow variation mode 0  
  Enables the value set by "SWeep Variation Logr"

- **Related program codes**

  SWeep Slow mode, SWeep Slow Target
3.4 Individual program code description

■ ?Error

Query:  ?Error
  (Answer)  ERROR <param>

- General:  Inquires error codes

- Answer message (when a query is run)
  - param:  Error code
    Format:  NR1, 3 characters

- Example setup
  ?error  Inquires an error code

- Remarks
  An error code will be output for the last error that occurred. If there is no error, then "0" will be output.
  For error codes and their descriptions, refer to "FRA5097 INSTRUCTION MANUAL".
  If this query is run, then the error code will be cleared.
3.4 Individual program code description

- **?IDentifier**

Query:  **?IDentifier**

(Answer)  IDENTIFIER <param>

- General: Inquires the model name

- Answer message (when a query is run)
  - **param**: Model name
    - Format: String "FRA5097"

- Example setup
  - ?identifier Inquires the model name
3.4 Individual program code description

## ?OVerload

**Query:**  ?OVerload  
(Answer)  OVERLOAD <param>

- **General:** Inquires the detection status of overload input

- **Answer message (when a query is run)**
  - **param:** Detection status of overload input
  - **Format:** NR1, 2 characters

<table>
<thead>
<tr>
<th>param</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No overload</td>
</tr>
<tr>
<td>1</td>
<td>Only CH1 is overloaded</td>
</tr>
<tr>
<td>2</td>
<td>Only CH2 is overloaded</td>
</tr>
<tr>
<td>3</td>
<td>CH1 and CH2 are overloaded</td>
</tr>
</tbody>
</table>

- **Example setup**
  - ?over  Inquires the overload status

- **Remarks**
  In response to a query using this program code, the overload status at the moment that the query has been received, will be returned. However, no previous overload input will be returned.
### SRqenable

**SRqenable**

**?SRqenable**

**Setup:**  `SRqenable <param>`

**Query:**  `?SRqenable`

(Answer)  `SROENABLE <param>`

- **General:** Sets or inquires the enabled status of SRQ transmission

- **Parameter (when parameters are set)**
  - **param:** Factors that enable SRQ transmission
    - Format: NR1  Range: 0 to 47

- **Answer message (when a query is run)**
  - **param:** Current factor that enables SRQ transmission
    - Format: NR1, 3 characters  Range: 0 to 47

- **Example setup**
  
  srq 4  
  Transmits an SRQ when overload input has occurred

- **Remarks**
  An SRQ will be transmitted when any of the following factors has taken place. Here a set value is assumed to be a binary code and the bit to be 1. The setting is made in decimal code. An SRQ transmission will be enabled when at least one of the following factors has occurred.

  - bit5(+32): Error occurred
  - bit4(+16): (Unused)
  - bit3(+8): Output ready (query, data transfer etc.)
  - bit2(+4): Overload input (overload detected)
  - bit1(+2): Measure end
  - bit0(+1): Sweep end

  For example, to transmit an SRQ when an error has occurred or a sweep ends, 32+1=33 should be set.
3.4 Individual program code description

- ?STatus

Query:  ?STatus
        (Answer)  STATUS <param>

- General: Outputs the status byte

- Answer message (when a query is run)
  - param: Status byte
    Format: NR1, 4 characters

- Example setup
  ?st  Reads the status byte

- Remarks
  The status byte identical to the serial poll byte will be output in decimal codes. However, bit3 (Output ready) will be constantly 0.
  For a description of the status byte, refer to "Table 2-1 Status Byte".
3.4 Individual program code description

■ ?Version

Query:  ?Version

   (Answer)   VERSION <param>

• General: Outputs the software version

• Answer message (when a query is run)
  • param: Software version
  Format: NR2. If the number of characters is less than 5, then they will be placed to left and a space(s) will fill in for the missing character(s).

• Example setup
  ?version

   (Example answer)   VERSION 1.00
3.4 Individual program code description

■ ?Learn

Query:  ?Learn
   (Answer)  <param> (block delimiter)
   <param> (block delimiter)
       :
   <param> (delimiter)

• General:  Output of all FRA5097 setting items

• Answer message (when a query is run)
  • param:  FRA5097 setup status
    Format:  String

• Example setup
  ?learn
   (Example answer)  CALCULATION ARITHMETIC  0, 0, 0, 0, 0,
                    CALCULATION JW   1, 0, 0
                       :

• Remarks
  Response messages are issued for all queries that have corresponding setup messages.  A header is applied to each parameter.  Responses are not issued for SETup Date or SETup Time.  The answer strings to be output will use the format that the FRA5097 can receive them as setup messages.
  For the list of setup messages, refer to "Table 3-2 Setup Message List".
4. Notes for Programming

a) If the FRA 5097 is designated as a talker without requesting any output (without sending any query message), then an empty block will be returned.

b) To perform a hold measurement using "Measure Repeat Off ; SWeep MEasure Hold" via the GPIB/USB and then read the data, follow these steps:
   1) Use "SRqenable 2 to enable the transmission of an SRQ (service request), which is intended for the end of a measurement.
   2) Start a hold measurement using "Measure Repeat Off ; SWeep MEasure Hold."
   3) Wait for RQS (bit6) and the end of measurement (bit1) to be set by means of a serial poll.
   4) Use "?DAta Read CUrrent" to inquire and read the data.

c) If the transmission of a program code etc. is interrupted using the controller before it is completed, an error may occur to the subsequent program code. When a program code transmission is interrupted in this manner, execute a device clear.

d) Assume that a query message is sent via the controller. The FRA 5097 is then designated as a talker and a data transfer is started. If the controller then interrupts the reception of data in progress, the FRA 5097 may freeze while waiting for a transmission. When a data transfer is interrupted in this manner, execute a device clear.

e) Assume that a query message is sent via the controller. Another query message is then sent without designating the FRA 5097 as a talker. In such a case, the answer to the latter query may directly follow that to the former query (without any delimiter being placed in between).
   In this manner, when a query message is sent that does not designate the FRA 5097 as a talker, execute a device clear.

g) If a query message is sent immediately after an error has occurred in the GPIB, a null character string may be returned. In such a case, execute a device clear after the error has occurred.
5. Sample Program

5.1 Overview of Sample Program.................................5-2
5.2 Setup and Query.........................................................5-3
5.3 Setting a Sweep Frequency Range and
   Sweep Measurement.....................................................5-5
5.4 Transfer of Measurement Data to Controller (1) ........5-7
5.5 Transfer of Measurement Data to Controller (2) ........5-9
5.1 Overview of Sample Program

This section describes some examples of remote controlling that uses the GPIB interface.

The equipment configuration used is as follows:

- **Computer:** IBM PC/AT compatible
- **OS:** Microsoft Windows 2000
- **Language:** Microsoft Visual Basic 6.0 or later
- **GPIB controller:** NF Corporation USB488 (USB-GPIB conversion adapter)

The description here will focus on the following four types of remote control:

a) **Setup and Query**
   This is the simplest type of remote control. After initialization, the amplitude of the oscillator is set, and then it is confirmed by running a query. In addition to GPIB, an example for USB is also shown.

b) **Setting a Sweep Frequency Range and Sweep Measurement**
   After the maximum and minimum sweep frequencies are set, a sweep measurement (Up Sweep) is started, and it is then waited until the sweep ends.

c) **Transfer of Measurement Data to Controller (1)**
   The measurement data obtained by a sweep measurement is transferred to the controller in ASCII format.

d) **Transfer of Measurement Data to Controller (2)**
   The measurement data obtained by a single measurement is transferred to the controller in ASCII format.

In each of these cases, an error check or a similar step is omitted. When you actually engage in programming, give consideration to the steps for error processing or initialization.
5.2 Setup and Query

When the button is pressed, the amplitude of the oscillator is set, and then the results of a query will be displayed.

The necessary initialization is performed with the Form_Load.

Here, the device descriptor (Dev) is opened with the timeout set to 300 ms, GPIB address to 2, EOI to ON, and terminator to LF. With the SetQueryBtn, a command for setting the amplitude of oscillator will be output to the device (FRA5097) using the character string that was entered in SetValueTxt, as a parameter.

With the Form_Unload, the equipment is put back in the local status.

### GPIB example

```vbnet
Const Adr As Integer = 2

Private Sub Form_Load()
    Dim stat As Integer
    initialize 0, 0
    transmit "DCL", stat
End Sub

Private Sub Form_Unload(Cancel As Integer)
    Dim stat As Integer
    transmit "UNL LSTEN " & CStr(Adr) & " GTL", stat 'Go to Local
End Sub

Private Sub SetQueryBtn_Click()
    Dim rdbuf As String
    Dim j As Integer
    Dim stat As Integer
    send Adr, "OScillator Amplitude " & SetValueTxt.Text, stat ' Setting of amplitude
    send Adr, "?OScillator Amplitude ", stat ' Query on amplitude
    enter rdbuf, 256, j, Adr, stat
    AnswerLbl = rdbuf ' Reception of answer message
End Sub
```
5.2 Setup and Query

USB example

Const serial As String = "0012045"  'FRA5097 serial No.
Const MAX_CNT = 200
Const idVendor As String = "0x0D4A::"  'NF corp.
Const idProduct As String = "0x000A::"  'FRA5097 product No.
Dim dfltrm As VSession
Dim sesn As VSession

Private Sub Form_Load()
    Dim stat As ViStatus
    Const timeout As Integer = 1000  'timeout=1000(ms)
    stat = viOpenDefaultRM(dfltrm)
    If (stat < VI_SUCCESS) Then
        Exit Sub
    End If
    stat = viOpen(dfltrm, "USB0::" & idVendor & idProduct & _
        serial & "::INSTR", VI_NULL, VI_NULL, sesn)
    stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, timeout)
    End Sub

Private Sub Form_Unload(Cancel As Integer)
    Dim stat As ViStatus
    stat = viGpibControlREN(sesn, VI_GPIB_REN_DEASSERT)
    stat = viClose(sesn)
    stat = viClose(dfltrm)
    End Sub

Private Sub SetQueryBtn_Click()
    Dim stat As ViStatus
    Dim retCount As Long
    Dim buffer As String * MAX_CNT
    buffer = "OScillator Amplitude " & SetValueTxt & vbCrLf
    stat = viWrite(sesn, buffer, Len(buffer), retCount)  ' Setting of amplitude
    wait (0.5)
    buffer = "?OScillator Amplitude" & vbCrLf
    stat = viWrite(sesn, buffer, Len(buffer), retCount)  ' Query on amplitude
    stat = viRead(sesn, buffer, MAX_CNT, retCount)  ' Reception of answer message
    AnswerLbl.Caption = Left$(buffer, retCount)
    End Sub

Private Sub wait(t As Double)
    Dim tm As Double
    tm = Timer
    While ((Timer - tm) < t)
        DoEvents
    Wend
    End Sub
5.3 Setting a Sweep Frequency Range and Sweep Measurement

When the button is pressed, a sweep frequency range is set and a sweep measurement will be started. Then, it will be waited until the sweep terminates.

A sweep range will be set if the button is pressed after entering the upper- and minimum sweep frequencies. When the status label (StsLbl) has switched from "Sweep in progress" to "Sweep End", then the sweep measurement is completed. If the set maximum or minimum sweep frequency is modified, then the status label (StsLbl) will be cleared. A sweep measurement in progress is identified by bit0 of the status byte that was obtained through a serial poll. To obtain the status byte, the timer function (Timer) of Visual Basic is used to execute a serial poll every two seconds. If a serial poll is performed without applying this weight, the measuring operation of the FRA5097 will become extremely slow, since it will cause the output of the status byte to be requested at frequent intervals.

If bit0 of the status byte becomes 1, the device will determine that the sweep is completed, and then the measurement will terminate after the status label is switched to "Sweep End".

(Contd.)
5.3 Setting a Sweep Frequency Range and Sweep Measurement

Const Adr As Integer = 2

Private Sub Form_Load()
    Dim stat As Integer
    initialize 0, 0
    transmit "DCL", stat
End Sub

Private Sub Form_Unload(Cancel As Integer)
    Dim stat As Integer
    transmit "UNL LSTEN " & CStr(Adr) & " GTL", stat 'Go to Local
End Sub

Private Sub SetSweepBtn_Click()
    Dim rdbuf As String
    Dim j As Integer
    Dim stat As Integer
    Dim stb As Integer
    Dim tm As Long
    StsLbl.Caption = "Sweep in Progress"
    DoEvents
    send Adr, "SWeep range " & SetSweepMin.Text & ", " & SetSweepMax.Text, stat
    send Adr, "SWeep MEasure Up", stat
    stb = 0
    tm = Timer
    While ((stb And &H1) = 0) ' Waits for bit0=1 of status byte
        While ((Timer - tm) < 2)
            tm = Timer
        Wend ' Waits two seconds
    spoll Adr, stb, stat ' Serial poll
    Wend
    StsLbl.Caption = "Sweep End"
End Sub

Private Sub SetSweepMax_Change()
    StsLbl.Caption = "" ' Status label cleared
End Sub

Private Sub SetSweepMin_Change()
    StsLbl.Caption = "" ' Status label cleared
End Sub
5.4 Transfer of Measurement Data to Controller (1)

When the button is pressed, the measurement data of the current tag in the FRA5097 will be loaded and displayed in ASCII format.

When the "READ" button is pressed, the current tag number of the FRA5097 will be loaded, followed by the data block count of the current tag. All data blocks of the current tag number will be read out when the data transfer format is set to ASCII and the data to be transferred is set to any of the following:

Frequency (Hz), Gain (dB), Phase (deg),

Each time one data block is sent, a delimiter (CR or CR+LF to be set in FRA5097) will be added, so that data loading will continue for as many times as the number of data blocks received.
Constr Adr As Integer = 2

Private Sub DataLoadBtn_Click()
    Dim rdbuf As String
    Dim j As Integer
    Dim stat As Integer
    Dim tag As Integer
    Dim datasize As Integer
    DataTxt.Text = ""
    DoEvents
    send Adr, "?DAta CUrrent", stat
    enter rdbuf, 256, j, Adr, stat
    tag = Val(rdbuf)  '  Current tag number
    send Adr, "?DAta Read Size", stat
    enter rdbuf, 256, j, Adr, stat
    datasize = Val(rdbuf) + 1 '  Block count of current tag
    send Adr, "DAta Template String, Sweep, LogR, Theta", stat
       '  Designation of transfer format
    send Adr, "?DAta Read data " & Str$(tag) & ",0, " & Str$(datasize - 1), stat
       '  Start of data transfer
    While (0 <= datasize)
        enter rdbuf, 256, j, Adr, stat  '  One block of data transfer by one round of reception
        DataTxt.Text = DataTxt.Text & rdbuf & Chr$(13) & Chr$(10)
        datasize = datasize - 1
    Wend
End Sub

Private Sub Form_Load()
    Dim stat As Integer
    initialize 0, 0
    transmit "DCL", stat
End Sub

Private Sub Form_Unload(Cancel As Integer)
    Dim stat As Integer
    transmit "UNL LSTEN " & CStr(Adr) & " GTL", stat 'Go to Local
End Sub
5.4 Transfer of Measurement Data to Controller (1)

5.5 Transfer of Measurement Data to Controller (2)
A single measurement will be carried out. Then, the measurement data will be loaded and displayed in ASCII format after the end of the measurement is waited.

When the "MEASURE" button is pressed, a single measurement will be started after the data transfer format and the data to be transferred are set. The measurement data will be loaded and displayed after the end of the measurement is confirmed by "?SWeep MEasure."

(Contd.)
Const adr As Integer = 2

Private Sub Command1_Click()
    Dim stat As Integer
    Dim rdbuf As String
    Dim j As Integer
    Dim tm As Long
    Dim sweep As Integer
    send adr, "DAta Template String,Sweep,LOGR,Theta", stat
    send adr, "SWeep MEasure Hold", stat ' Start of SINGLE Measurement
    ' Waits for measurement to end
    tm = Timer
    sweep = 1
    While (sweep <> 0)
        While (Abs(Timer - tm) < 1)
            send adr, "?SWeep MEasure", stat
            enter rdbuf, 256, j, adr, stat
            sweep = Val(rdbuf)
            Wend
            send adr, "?DAta Read CUrrent", stat ' Query on data
            enter rdbuf, 256, j, adr, stat ' Reception of measurement data
            Text3.Text = Left$(rdbuf, 17)
            Text1.Text = Mid$(rdbuf, 19, 8)
            Text2.Text = Mid$(rdbuf, 28)
        Wend
    End Sub

Private Sub Form_Load()
    Dim stat As Integer
    initialize 0, 0
    transmit "DCL", stat
End Sub

Private Sub Form_Unload(Cancel As Integer)
    Dim stat As Integer
    transmit "UNL LSTEN" & CStr(adr) & "GTL", stat 'GTL
End Sub
6. Specifications

6.1 Interface Functions .............................................6-2
6.2 GPIB Bus Driver .................................................6-2
6.3 Applied Codes ....................................................6-2
6.4 Answers to Interface Messages ........................ 6-3
6.5 Multi-line Interface Messages .........................6-4
6.1 Interface Functions

### Interface Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Subset</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source handshake</td>
<td>SH1</td>
<td>All send handshake functions available</td>
</tr>
<tr>
<td>Acceptor handshake</td>
<td>AH1</td>
<td>All receive handshake functions available</td>
</tr>
<tr>
<td>Talker T6</td>
<td></td>
<td>Basic talker functions; and talker cancellation by MLA</td>
</tr>
<tr>
<td>Listener L4</td>
<td></td>
<td>Basic listener functions; and listener cancellation by MTA</td>
</tr>
<tr>
<td>Service request SR1</td>
<td></td>
<td>All service request functions available</td>
</tr>
<tr>
<td>Remote/local RL1</td>
<td></td>
<td>All remote and local functions available</td>
</tr>
<tr>
<td>Parallel poll PP0</td>
<td></td>
<td>No parallel poll function available</td>
</tr>
<tr>
<td>Device clear DC1</td>
<td></td>
<td>All device clear functions available</td>
</tr>
<tr>
<td>Device trigger DT0</td>
<td></td>
<td>No device trigger function available</td>
</tr>
<tr>
<td>Controller C0</td>
<td></td>
<td>No controller function available</td>
</tr>
</tbody>
</table>

6.2 GPIB Bus Driver

### Bus Driver Specification

<table>
<thead>
<tr>
<th>Data bus</th>
<th>DIO1 to 8</th>
<th>Open collector</th>
</tr>
</thead>
<tbody>
<tr>
<td>Handshake bus</td>
<td>NRFD, NDAC, DAV</td>
<td>Open collector</td>
</tr>
<tr>
<td>Control bus</td>
<td>SRQ, EOI</td>
<td>Open collector</td>
</tr>
</tbody>
</table>

6.3 Applied Codes

When designated as a listener, the FRA5097 will accept ISO 7bit codes (JIS/ASCII) except for the reception of binary format data, and will ignore any MSB of 8bit data (parity etc.). Program codes are case-independent, so that a program code either in uppercase or lowercase will be interpreted and executed identically. Control characters will be ignored, except <CR>, <LF> and <TAB> (hexadecimal codes: 0 to 8, B, D to 1F, and 7F).

When designated as a talker, the FRA5097 will accept ISO 7bit codes (JIS/ASCII) without parity (8bit data MSB = 0), except for the transmission of binary format data. All alphabetical symbols in a program code will be transmitted in uppercase. When sending binary format data, the FRA5097 will accept all 8bit patterns and use the standard IEEE floating-point format (including all control codes). Refer to "3.2 Data Transfer" for information on floating-point formats.
### 6.4 Answers to Interface Messages

#### Answers to Interface Messages

<table>
<thead>
<tr>
<th></th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IFC</strong></td>
<td>· Initializes the interface.</td>
</tr>
<tr>
<td></td>
<td>· Resets the listener/talker designation.</td>
</tr>
<tr>
<td><strong>DCL and SDC</strong></td>
<td>· Clears the I/O buffer.</td>
</tr>
<tr>
<td></td>
<td>· Clears an error.</td>
</tr>
<tr>
<td></td>
<td>· Resets the transmission of SRQ, and resets the factor bit in status byte.</td>
</tr>
<tr>
<td></td>
<td>· Disables the transmission of SRQ.</td>
</tr>
<tr>
<td></td>
<td>· Disables the header/mnemonic output.</td>
</tr>
<tr>
<td></td>
<td>· Resets the data I/O format to default.</td>
</tr>
<tr>
<td><strong>LLO</strong></td>
<td>· Disables the LOCAL key on the panel.</td>
</tr>
<tr>
<td><strong>GTL</strong></td>
<td>· Changes to the local status.</td>
</tr>
</tbody>
</table>
### 6.5 Multi-line Interface Messages

#### Table

<table>
<thead>
<tr>
<th>Column</th>
<th>ROW</th>
<th>b4</th>
<th>b3</th>
<th>b2</th>
<th>b1</th>
<th>b6</th>
<th>b7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Address</th>
<th>Command Group (ACG)</th>
<th>Universal Command Group (UCG)</th>
<th>Listener Address Group (LAG)</th>
<th>Talker Address Group (TAG)</th>
</tr>
</thead>
</table>

#### Notes

- **"1** MSG is an interface message
- **"2** b1=DIO1 · · · b7=DIO7. DIO8 is not used
- **"3** Involves a secondary command
- **"4** “\" in IEC standard; “¥” in JIS

---

**Address**: Universal Command Group (UCG) | **Listener Address Group (LAG)** | **Talker Address Group (TAG)**

**Primary Command Group (PCG)**

**Secondary Command Group (SCG)**

- **TCT**: Take Control
- **LLO**: Local Lockout
- **DCL**: Device Clear
- **PPU**: Parallel Poll Unconfigure
- **GET**: Group Execute Trigger
- **SDC**: Selected Device Clear
- **PPC**: Parallel Poll Configure
- **UNL**: Unlisten
- **UNT**: Untalk

---

FRA5097 GPIB/USB 6-4
If there are any misplaced or missing pages, we will replace the manual. Contact the sales representative.

NOTES:

- Reproduction of the contents of this manual is forbidden by applicable laws.
- The contents of this manual may be revised without notice.
- Information provided in this manual is intended to be accurate and reliable. However, we assume no responsibility for any damage regarding the contents of this manual.
- We assume no responsibility for influences resulting from the operations in this manual.

Copyright 2006 - 2010, NF CORPORATION

FRA5097 Frequency Response Analyzer GPIB/USB Instruction Manual

NF CORPORATION
3-20 Tsunashima Higashi 6-chome, Kohoku-ku. Yokohama-shi
223-8508, JAPAN
Phone: 81-45-545-8128 Fax: 81-45-545-8187